全相位数字信号
处理方法及MATLAB实现

苏飞　张楷亮　曹继华　著



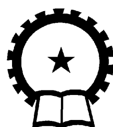
机械工业出版社
CHINA MACHINE PRESS

VISIT…

LANZAROTE
Caliente.com

全相位数字信号处理方法及MATLAB实现

苏　飞　张楷亮　曹继华　著



机 械 工 业 出 版 社

本书系统地介绍了全相位处理数字信号的系统组成、基本概念、基本原理、分析和设计方法。全书共10章，内容包括：绪论、全相位理论基础、基于全相位的窗函数设计、正交域全相位变换、二维全相位变换、基于W-O的谱分析、设计滤波器组的全相位方法、设计模板的全相位方法、基于W-O的自适应信号处理、全相位处理的Simulink仿真。本书中的理论证明过程简明扼要，注重设计实践，全部实验均通过MATLAB进行算法设计和Simulink仿真验证，且提供全部源码程序。

本书可作为高等学校工科电子信息工程、计算机科学与技术、信号与信息处理等类各专业研究生参考资料，也适合高年级本科生阅读，也可作为电子与电气、自动控制类科研人员和工程技术人员的参考书。

图书在版编目（CIP）数据

全相位数字信号处理方法及MATLAB实现/苏飞，张楷亮，曹继华著．—北京：机械工业出版社，2014.12

ISBN 978-7-111-48233-8

Ⅰ.①全…　Ⅱ.①苏…②张…③曹…　Ⅲ.①Matlab软件—应用—数字信号处理　Ⅳ.①TN911.72

中国版本图书馆CIP数据核字（2014）第234817号

机械工业出版社（北京市百万庄大街22号　邮政编码100037）
策划编辑：任　鑫　责任编辑：任　鑫
版式设计：赵颖喆　责任校对：闫玥红
封面设计：路恩中　责任印制：李　洋
三河市宏达印刷有限公司印刷
2015年1月第1版第1次印刷
169mm×239mm·23印张·440千字
0001—3000册
标准书号：ISBN 978-7-111-48233-8
定价：59.80元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

电话服务	网络服务
社服务中心：（010）88361066	教材网：http://www.cmpedu.com
销售一部：（010）68326294	机工官网：http://www.cmpbook.com
销售二部：（010）88379649	机工官博：http://weibo.com/cmp1952
读者购书热线：（010）88379203	**封面无防伪标均为盗版**

前　言

随着数字信号处理器件DSP、FPGA等运算速度的不断提高，在各个应用领域，研究者们对处理质量的要求也越来越高。如何降低数字信号截断效应的影响是无法回避的问题，通过提高采样率和采用重叠处理方法在一定程度上使处理效果得到了改善，但对存储空间大小提出了较高的要求。全相位法是在重叠方法基础上提出的新型数字信号处理方法，它使输入数据按照设计需求进行不同加权后，以中心数据作延拓重复处理。由于输入数据逐点移动而产生每个输出点，因此，也形象地称之为“全相位”。

全相位数据处理方法从20世纪80年代初提出以来，在滤波器设计、窗函数设计、谱分析、自适应信号处理、模板设计和数据内插、数据压缩方面的应用逐渐得到挖掘。在滤波器设计方面，采用全相位法使传输序列*H*的物理意义更明确（即调整*H*可直接控制各个频率分量），更方便设计基于DFT、DCT和DWT等各种正交变换的数字滤波器，且具有极低的旁瓣衰减。本书在一维全相位滤波器设计的基础上，首次实现了二维全相位处理结构及理论分析；在窗函数设计方面，采用LMS方法得到的优化窗的过渡带低于传统方法，采用矩形二叉分解实现了各种衰减指数的似余弦窗。本书在此基础上，结合全相位传输函数算术表达式，创造性地提出了最小二乘窗函数设计法，得到的基窗消除了在通带和阻带内的波纹。在谱分析方面，全相位法被证明可以提高主瓣能量，且可有效抑制旁瓣，即提高了信号检测能力。此外，全相位FFT还具有“相位不变性”（即能量最大谱线决定相位）。在自适应信号处理方面，结合全相位处理思想的LMS算法从理论上得以证明，其收敛速度和稳态误差均优于传统算法，在自适应信号辨识、自适应除噪等方面得到实验验证。在模板设计方面，则充分利用了加窗重叠的全相位实质，分别在DFT、DCT和DWT三个变换域实现二维模板的设计并应用到图像数据内插方面，通过与传统的线性、小波等方法对比，全相位模板对于中低频图像表现出了更好的数据平滑特性。

全相位数字信号处理的基本概念和理论基础是由天津大学王兆华教授和侯正信教授在20世纪80年代初期奠定的。最早源于王兆华教授提出的将包含某像素的所有分块都进行考虑计算的重叠数字滤波方法，并以此方法为基础建立了重叠内插模板。侯正信教授在此基础上，首先将重叠滤波模板的构造从DFT域、

Walsh 域延伸到 IDCT 域并提出了离散列率滤波器的概念。其次，侯正信教授在深刻认识此最大程度重叠处理方法的本质后，把此遍历所有位置时刻的重叠正名为“全相位”。自 2001 年以来，王兆华教授和侯正信教授指导多名研究生不断丰富全相位滤波的理论，并挖掘在多个领域的应用。目前，已经涉及的应用方面有自适应信号处理、窗函数设计、滤波器组设计、图像内插、数据内插等。

本书在汇总全相位数字滤波器设计、谱分析、自适应信号处理和模板设计的基础上，融入二维全相位信号处理的实现结构、最小二乘窗函数设计以及多级全相位 DSP 仿真系统等几方面新的研究成果。本书的全部理论推导都是通过 MATLAB 进行验证的，并设计了相关实验对比测试，提供了大量算法及源码，具有较大的参考价值。

参加本书编写的还有天津理工大学张楷亮教授和天津职业技术师范大学曹继华教授。此外，本书在出版过程中还得到了天津理工大学吕联荣副教授、卢晋副教授以及图书馆张斌主任的肯定与支持，天津师范大学徐妮妮副教授也给予了宝贵的建议。这里一并向各位表示感谢和敬意!

限于著者的水平，书中难免有不妥和错误之处，恳请读者批评指正。

苏　　飞

2014 年 7 月

目　录

前言
第 1 章　绪论 …… 1
1.1　全相位滤波方法 …… 1
1.1.1　截断引入的误差 …… 1
1.1.2　FIR 滤波器的频率泄漏 …… 3
1.1.3　图像分块处理造成的马赛克现象 …… 4
1.1.4　全相位法解决截断误差 …… 4
1.2　全相位滤波理论的发展现状 …… 6
1.3　MATLAB 在信号处理中的应用 …… 7
1.4　本书的主要内容 …… 9
1.5　本书的主要特点 …… 10
第 2 章　全相位理论基础 …… 11
2.1　FIR 数字滤波器设计方法 …… 11
2.1.1　常规法 …… 12
2.1.2　窗函数法 …… 19
2.1.3　频率抽样法 …… 23
2.1.4　切比雪夫逼近法 …… 25
2.1.5　其他方法 …… 26
2.2　全相位数字滤波器 …… 26
2.2.1　DFT 域无窗全相位数字滤波器 …… 28
2.2.2　DFT 域单窗全相位数字滤波器 …… 35
2.2.3　DFT 域双窗全相位数字滤波器 …… 40
2.3　全相位数字滤波器特性 …… 45
2.4　应用举例 …… 61
2.4.1　基于 ap 的信号滤波 …… 61
2.4.2　边界拓展 …… 64
2.4.3　噪声多带分解 …… 67
2.5　小结 …… 69
第 3 章　基于全相位的窗函数设计 …… 70
3.1　滤波器设计中的窗函数 …… 70
3.1.1　窗的作用 …… 70
3.1.2　传统窗特性 …… 75
3.1.3　apDW 频率取样误差分析 …… 87
3.2　基于 LMS 准则的基窗设计算法 …… 91
3.3　基于矩形二叉分解的基窗设计 …… 97
3.4　最小二乘基窗设计算法 …… 102
3.5　小结 …… 105
第 4 章　正交域全相位变换 …… 106
4.1　正交变换域 …… 106
4.1.1　一维变换的正交基 …… 106
4.1.2　沃尔什正交基 …… 107
4.1.3　余弦正交基 …… 111
4.1.4　其他正交变换 …… 114
4.1.4.1　哈达玛变换 …… 114
4.1.4.2　数论变换 …… 116

4.1.4.3 卡-洛变换 ………… 117
4.2 变换域全相位滤波器设计 ………… 119
4.2.1 DCT 域全相位滤波器 ………… 119
4.2.2 DWT 域全相位滤波器 … 124
4.2.3 变换域全相位信号处理 … 126
4.3 小结 ………… 133
第 5 章 二维全相位变换 ………… 134
5.1 二维线性系统 ………… 134
5.2 二维变换正交基 ………… 138
5.2.1 二维傅里叶变换 ………… 140
5.2.2 二维沃尔什和哈达玛变换 ………… 145
5.2.3 二维正弦变换 ………… 150
5.3 二维全相位信号处理 ………… 151
5.4 二维 DFT 全相位 ………… 156
5.5 二维 DCT/DWT 全相位 … 170
5.6 二维全相位信号处理特性 ………… 176
5.7 小结 ………… 186
第 6 章 基于 W-O 的谱分析 … 187
6.1 谱估计的一般理论 ………… 187
6.1.1 经典功率谱估计 ………… 188
6.1.1.1 直接法 ………… 188
6.1.1.2 间接法 ………… 188
6.1.1.3 直接法的改进 …… 188
6.1.2 现代功率谱估计 ………… 193
6.1.2.1 参数模型法 ………… 193
6.1.2.2 谱估计存在的问题 … 196
6.2 全相位谱估计 ………… 197
6.2.1 基于 W-O 算法的谱估计 ………… 197
6.2.2 W-O 谱分析在信号检测中的应用 ………… 203
6.3 小结 ………… 205
第 7 章 设计滤波器组的全相位方法 ………… 206
7.1 抽取和内插 ………… 206
7.1.1 一维信号分析 ………… 206
7.1.2 二维信号分析 ………… 211
7.2 滤波器组 ………… 216
7.2.1 双通道正交镜像滤波器组（2QMFB） ………… 216
7.2.2 2QMFB 完全重建系统 … 219
7.2.3 全相位半带滤波器设计 … 220
7.2.4 全相位半带滤波器的谱分解 ………… 229
7.3 图像子带分解 ………… 240
7.4 全相位 2QMF 多分辨率实验 ………… 243
7.4.1 一维信号分解与重构 …… 243
7.4.2 二维信号分解与重构 …… 249
7.5 小结 ………… 254
第 8 章 设计模板的全相位方法 ………… 255
8.1 模板设计理论 ………… 256
8.1.1 基于二维沃尔什变换内插模板 ………… 257
8.1.2 基于二维傅里叶变换内插模板 ………… 261
8.1.3 基于二维 DCT 变换内插模板 ………… 263
8.2 亚奈取样图像的二维谱分析 ………… 266
8.2.1 二维列率谱分析 ………… 266
8.2.2 二维频谱分析 ………… 268
8.3 全相位内插模板设计 …… 269
8.3.1 沃尔什内插模板 ………… 269
8.3.2 傅里叶内插模板 ………… 272
8.4 传统插值算法 ………… 273

8.4.1　理想内插 …………………… 274
8.4.2　最近邻域插值 ……………… 275
8.4.3　双线性内插 ………………… 275
8.4.4　B样条内插 ………………… 276
8.4.5　立方卷积内插 ……………… 279
8.4.6　小波内插 …………………… 280
8.5　全相位插值算法……………… 285
8.5.1　Ⅰ型全相位插值 …………… 285
8.5.2　Ⅱ型全相位插值 …………… 289
8.6　图像内插实验………………… 295
8.7　小结…………………………… 301
第9章　基于W-O的自适应信号处理 …………………… 303
9.1　引言…………………………… 303
9.2　自适应滤波器概述…………… 304
9.2.1　最佳滤波准则 ……………… 304
9.2.2　自适应滤波器的应用……… 304
9.2.3　自适应滤波算法 …………… 305
9.3　传统自适应算法……………… 306
9.3.1　最小均方滤波算法 ………… 306
9.3.2　最陡下降法 ………………… 308
9.3.3　最小均方LMS算法 ……… 309
9.4　时域加权重叠LMS算法（WO-LMS） …………………… 313
9.5　变换域加权重叠LMS算法（WO-TLMS） ……………… 317
9.6　WO-LMS在自适应辨识中的应用……………………… 319
9.7　WO-LMS在自适应除噪中的应用……………………… 329
9.8　小结…………………………… 334
第10章　全相位处理的Simulink仿真 ………………………… 336
10.1　MATLAB简介 …………… 336
10.1.1　数值计算 ………………… 337
10.1.2　图形功能 ………………… 337
10.1.3　工具箱 …………………… 338
10.2　Simulink仿真 …………… 340
10.2.1　Simulink简介 …………… 341
10.2.2　Simulink中的白噪声分析 ………………………… 343
10.2.3　基于小波的数字信号处理 ………………………… 343
10.3　全相位滤波器仿真 ……… 346
10.4　小结 ……………………… 350
附录 …………………………………… 351
附录A　矩阵谱证明 …………… 351
附录B　多相分解 ……………… 353
附录C　几种特殊的滤波器 …… 356
参考文献 ……………………………… 358

第1章 绪论

1.1 全相位滤波方法

数字信号处理是根据期望的结果而选用某种特定的数字计算方式对信号进行处理和加工的过程，一般称之为 DSP（Digtial Signal Processing）。这里的信号包括一维信号（如电压、电流等时域取样信号），也可以是二维信号（如静止图像矩阵），三维信号（如 2D 视频序列）及四维信号（如 3D 视频序列）等。针对不同类型的信号，信号处理涉及的内容主要包含滤波、变换、检测、谱分析、估计、压缩、识别等方面。滤波的主要任务是设计特性数字滤波器以得到不同频率成分的信号分量，如高通滤波可得到信号的高频成分，中值滤波则可降低高频分量起到平滑的效果。变换的主要任务是找到去相关性的基矩阵以把信号映射到基矩阵平面上来进行分析，如小波变换提供了时间和频率的双重变换，可有效地对信号进行多分辨率分析，傅里叶变换则将信号变换到频域上而得到不同的频率成分。此外，还有用于数据压缩的 DCT 变换、算术变换及用于通信领域的同步正交沃尔什变换等。检测和估计主要是从受噪声干扰的杂波中利用信号的统计特性得到有用信号的过程。谱分析则是从频域的角度对信号进行分析，尤其对平稳的随机信号，通过其稳定的相关函数可以直接得到功率谱分布特性。压缩是信号处理的一个重要方面，它主要是消除信号自身的相关性（如时间相关、空间相关及统计相关）从而达到减少数据量的效果。识别是通过提取信号某些关键的特征量后映射到对象数据库中进行匹配的过程，如语音识别、脸型识别等，其在智能物联网中的应用越来越广泛。

1.1.1 截断引入的误差

不管是信号处理的哪种应用，都是对输入信号进行某种特定的处理后得到期望的结果。从这个角度来理解，数字信号处理的过程就是一种滤波的过程，不同的应用只是采用滤波的方法不同而已。但有一个必须正视的问题，就是数字信号是通过对原始模拟信号按照奈奎斯特取样定理而得到，在实际处理时由于受到物理设备条件的限制，往往是对有限长度的数字信号进行分段处理，从而引入截断

误差。如理想矩形波采用有限阶数的正弦函数表示时会出现吉布斯振荡，物理可实现的有限长度滤波器在过渡带两边也会出现振荡现象。很明显，对于非周期信号局部的特性与整体特性是有差异的，如白噪声信号在截取后的均值、方差等各统计特性都会发生变化，而通过分析截断信号的特性去理解原始信号就会引入误差。对于周期信号也是如此，若取样频率不是信号频率的整数两倍以上，则取样信号的特性将发生改变。我们通过下面这个例子证明：分别以 $f_{s1}=128\text{Hz}$ 和 $f_{s2}=135\text{Hz}$ 的采样频率对单频正弦波 $x(t)=\sin(2\pi f_0 t)$ 进行采样后得到的序列为 $x_1(n)=\sin(2\pi f_0/f_{s1}n)$ 和 $x_2(n)=\sin(2\pi f_0/f_{s2}n)$。假设 $f_0=16\text{Hz}$，FFT 变换后的幅频特性如图 1-1 所示。

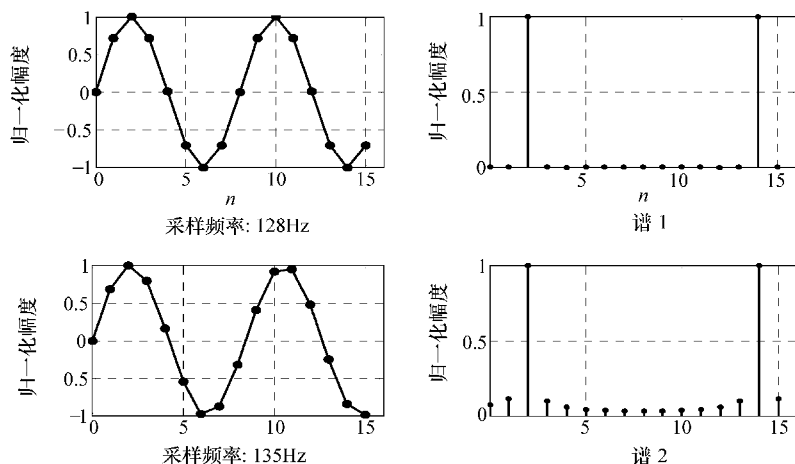


图 1-1 两种采样频率下频谱对比

由于采样频率 135Hz 不是 16Hz 整数倍，频谱出现扩散且扩散程度与采样频率偏离程度有关。图 1-2 给出了扩散程度对于采样频率与 128Hz 的偏离度的规律。在整倍频和半整倍频之间的采样会带来幅频偏差，且会随着频率偏移的增大而增大。

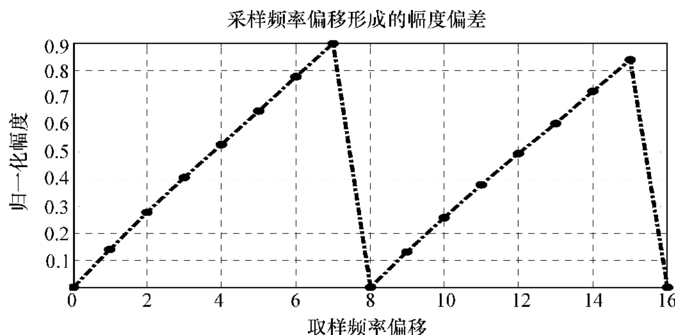


图 1-2 频率偏离带来的幅频偏差

1.1.2 FIR 滤波器的频率泄漏

利用有限长度的单位冲击响应序列对输入信号处理的过程（即根据一个周期内相位的不同）给予不同的加权就是 FIR 滤波器的基本原理。对应频域内是用有限个位置的采样点来拟合系统特性，在通带、阻带及其之间的过渡区间会出现振荡，这就是吉布斯现象，这正是由单位冲击响应 $h(n)$ 截断引起的。通过增大阶数 N 来改善频率泄漏，但波纹数量并没有减少，其带来的最大好处是过渡带宽度降低。如图 1-3 所示为阶数分别为 5、15、25 时按照矩形窗窗函数法设计的 3 种 FIR 滤波器特性曲线。

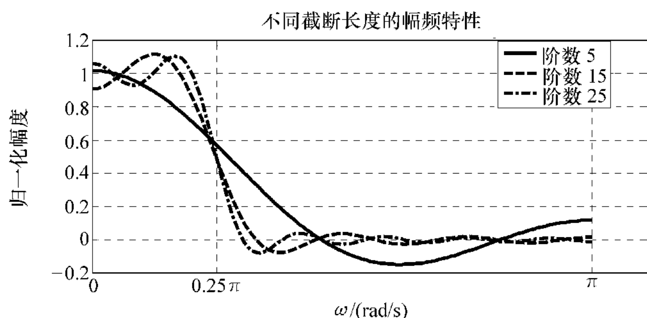


图 1-3 3 种截断长度的 FIR 滤波器幅频特性

为减小吉布斯现象，可选取一些旁瓣较小的窗口，如三角窗、汉宁窗等来代替矩形窗。图 1-4 中给出了 $N=20$ 时分别加矩形窗和余弦窗后的幅频特性曲线。

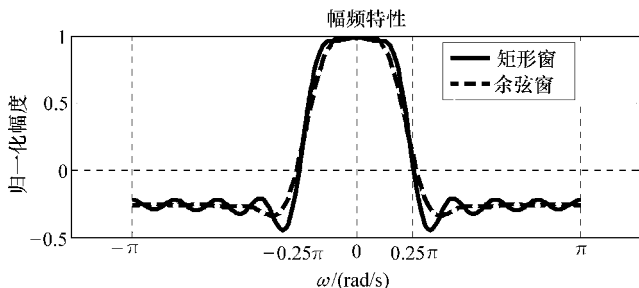


图 1-4 3 种截断长度的 FIR 滤波器幅频特性

使用余弦窗后，通带内的振荡基本消失，阻带内的波纹也大大减小，滤波器的性能得到了改善。但是，这是以过渡带的加宽为代价的，在同样阶数的情况下，采用余弦窗后过渡带宽度将增加了 35%。

1.1.3 图像分块处理造成的马赛克现象

MPEG-I/II、H.264 与 AVS 压缩编码标准中对视频序列的图像处理采用分块的方式来进行，即分成 8×8 的块后分别在本块内进行 DCT 变换、量化、滤波及反变换，然后再重新拼接。之所以选择 8 个像素是因为图像的相关距离一般不超过 8 个像素。经过变换以后，图像块的变换系数矩阵内部已消除了大部分的相关性，但是相邻的矩阵之间还是有一定的相关性，分块处理则不会考虑块间的关联性。因此，拼接后在块的边界会出现不连续的“锯齿”现象(即马赛克)。下面以 245×256 图像为例进行说明。整体和分块进行 DCT 变换再进行低通滤波后的效果图对比如图 1-5 所示。

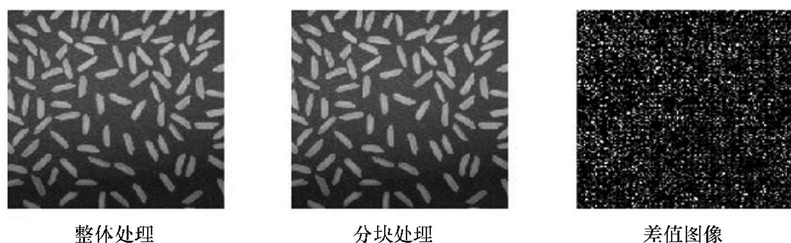


图 1-5 分块处理带来的马赛克

在差值图像中明显可以看到边界的不连续，这正是由分块处理带来的马赛克。

1.1.4 全相位法解决截断误差

为降低或解决截断误差，国内外有些专家和学者提出了一些改善措施。对截断信号进行“加窗”是一种常用的方法。以 FIR 滤波器设计为例，有限长度低通滤波器等于理想低通滤波器与矩形窗的乘积，频域则为差值函数与矩形窗频谱的卷积。

由图 1-6 可以看出，调整窗口长度 N 可以有效地控制过渡带宽度，但是波动幅度没有得到改善。而选择不同形状的窗函数使其谱函数的主瓣包含更多的能量，相应旁瓣的幅度就会减少，而旁瓣的减少则可使通带、阻带波动减少，从而加大阻带衰减，但这是以增大过渡带为代价的。

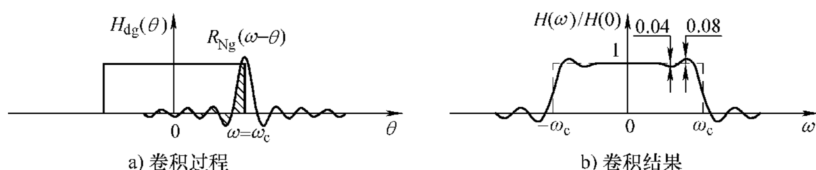


图 1-6 理想低通滤波器加矩形窗

重叠法是由 Malvar 等人在 DCT 基础上提出的解决“方块效应”的方法。普通的块变换在把输入信号分块后，块与块之间相互独立地进行变换运算，而且其变换矩阵是方阵，即输出数与输入数相等。而在重叠式变换中，信号在被分块之后，每块信号需要和相邻块的部分信号一起做变换运算，因此其变换矩阵的输出数 M 和输入数 L 一般不等，通常的取法是使 $L=2M$，即进行 50% 重叠处理。对于大多数图像，双正交重叠变换的压缩效果甚至好于 Daubechies9/7 小波。用 DFT 计算线性卷积中的重叠保留法是重叠算法的另一个应用。它将相临数据段的数据重叠。设输入数据段长度为 N_1，系统响应长度为 N_2。每个相继的输入段都由N_1-N_2+1 个新点与前一段保留下来的 N_2-1 个点组成。在每次运算得到的 N_1+N_2-1 个点中只有后面 N_2 个点的结果正确，而舍去结果的前 N_1-1 点。数据块的任何重叠率都是可行的，而其中以 50% 的重叠效率最高。重叠算法另一个重要应用是图像的二维重叠滤波。用常规的滤波器对图像处理时，在块的边缘会出现方块锯齿效应，虽然可以通过加大计算矩阵来减小锯齿，但这样就加大了计算量，同时锯齿效应仍然存在。这是由于进行二维离散变换时所取的方块尺寸总是有限的，用许多有限方块分别做处理后拼成一个整幅图像总会有边缘效应。二维重叠滤波将传统方法重叠进行，在原来处理矩阵的基础上向左右或上下移动一位再分别滤波，最后的输出是这些重叠滤波的叠加。由于向不同方向移动后块的高频误差是相反的，因此重叠后斜线处的方块锯齿效应消失。

全相位滤波法则是使用最大重叠的新方法，它考虑了对某个输入样本的所有长度为 N 的分段情况，如图 1-7 所示。

$x(n)$	$x(n-1)$	$\cdots$	$x(n-N+1)$	$\rightarrow$	$y^{(0)}(n)$
$x(n+1)$	$x(n)$	$\cdots$	$x(n-N+2)$	$\rightarrow$	$y^{(1)}(n)$
$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\vdots$
$x(n+N-1)$	$x(n+N-2)$	$\cdots$	$x(n)$	$\rightarrow$	$y^{(N-1)}(n)$

图 1-7　全相位滤波的输入对应的所有输出

假定分段信号的长度为 N，则其中任何一个输入点 $x(n)$ 都可以出现在 N 个数据段中，$x(n)$ 在第 k 个数据段中的位置是 k，因此，$x(n)$ 将遍历数据段中的所有时刻。第 k 段输入所对应 $x(n)$ 的输出是 $x(n)$ 在输入段中第 k 个位置时的结果，将每段 N 个输出结果中对应 $x(n)$ 的输出算术平均作为系统输入为 $x(n)$ 时的输出，这就是全相位滤波的基本思想。

显然，N 阶全相位滤波中的输入段是采用 $(N-1)/N$ 重叠的，即最大程度重叠。这样，对应数据的连续输入将产生连续的输出，且消除了常规方法中由于分段所产生的截断效应。它的另一个突出特点是便于对信号频率成分进行控制，对于满足一定条件的全相位滤波器，可实现完全线性相位，幅频特性则表现为通带

和阻带波纹消失且衰减增大。这里所说的“全相”的概念是对时刻而言，因为滤波器的输入在遍历所有时刻后才得到相应的输出。

1.2 全相位滤波理论的发展现状

全相位数字信号处理的基本概念和理论基础是由天津大学王兆华教授和侯正信教授在 20 世纪 80 年代初期奠定的。最早源于王兆华先生提出的将包含某像素的所有分块都进行考虑计算的重叠数字滤波方法，并以此方法为基础建立了重叠内插模板，在 1/2、1/3、1/4、1/8 抽取和内插及图像放大方面得到应用。2001 年，王兆华教授发表了 DFT 域的无窗重叠数字滤波器的框图，并指出一维重叠数字滤波可等效为 FIR 滤波器处理。在 2002 ~ 2004 年，王兆华教授申请了 3 项相关全相位的发明专利，即“一种频域自适应滤波器”、“一种带窗的频域频率器”和“全相位 FFT 频域分析装置”，并指出全相位数字滤波器的传输特性序列 $\boldsymbol{H}$ 具有与连续信号系统特性相似整体可控特点。王兆华教授首次给出了无窗、单窗和双窗全相位滤波器的结构形式及其频率特性图；其提出的“全相位 FFT 谱分析”极大地抑制了频谱泄漏，所有这些奠定了全相位滤波的理论基础。侯正信教授在此基础上，首先将重叠滤波模板的构造从 DFT 域、沃尔什域延伸到 IDCT 域并提出离散列率滤波器的概念；之后，侯正信教授在深刻认识此最大程度重叠处理方法的本质后，把此遍历所有位置时刻的重叠正名为“全相位”。

至 2001 年以来，王兆华教授和侯正信教授指导多名研究生不断丰富全相位滤波的理论并挖掘在多个领域的应用。目前，已经涉及的应用方面有自适应信号处理、窗函数设计、滤波器组设计、图像内插、数据内插等。在自适应信号处理方面，苏飞博士把全相位滤波的数据预处理与 LMS 方法相结合，并运用到了自适应除噪、自适应系统辨识中，其收敛速度和稳态误差指标均优于传统方法；黄翔东博士从全相位预处理数据的统计特性、自相关特性给予了证明。在窗函数设计方面，苏飞博士从全相位滤波器结构出发，推导出窗函数的重要特性的几个方面，并提出“基于最小二叉”和“最小均方误差迭代”两种设计窗函数的新方法。在滤波器组设计方面，苏飞博士通过设置系统传输特性序列 $\boldsymbol{H}$ 设计得到了严格功率互补的滤波器组，并应用到图像的多分辨率分析中，且与小波分析结果进行了对比。徐妮妮博士则利用无窗全相位滤波器频率特性非负的特点，结合谱分解思想设计出了全相位半带滤波器并应用到静止图像的压缩编码中。黄翔东博士和黄晓红博士在全相位 FFT 谱分析的理论基础上，全面挖掘了在频谱方向上的应用（如频谱校正等）。此外，两人在各种滤波器设计方面也取得大量研究成果，如间断特性滤波器、对称滤波器和零相位滤波器。

随着全相位滤波理论的不断丰富和应用扩展，越来越多的学者开始认识和使

用全相位滤波方法。如江苏科技大学电信学院的黄凯华在《基于加窗插值全相位的谐波检测方法》中将全相位滤波理论的相位不变特性应用到了电网谐波中极大地提高了检测谐波的频率、幅度和相位；西安邮电学院的王松在《基于全相位 FFT 的位码捕获研究》中阐述了基于全相位 FFT（apFFT）设计的一种变换域伪码捕获方法，可完成对多普勒频高精度估计，并且该方法具有更强的抗噪性能；天津大学精仪学院杨颖在《全相位谱分析在自混合干涉位移测量中的应用》中通过应用全相位滤波方法，得到在不增加光学器件的条件下位移测量误差降到纳米级以下的优良结果。此外，全相位滤波理论也在通信、语音信号处理、信号重构等方面得到了应用。

在网上，对全相位感兴趣的学者提出了一些问题，如“对王兆华老师提出的全相位滤波器，如果要对实际采样得到的信号进行直接运算的话，不能得到其真实的初始相位，但复制和频率的校正结果还是很好的”、“如何对带窗全相位数字滤波器的窗进行归一化”等在实际应用中采用 MATLAB 编码中遇到的具体问题。

1.3　MATLAB 在信号处理中的应用

MATLAB 是美国 MathWorks 公司出品的商业数学软件，用于算法开发、数据可视化、数据分析以及数值计算的高级技术计算语言和交互式环境，主要包括 MATLAB 和 Simulink 两大部分。其包含数字信号处理在内多个专业工具，为工程师测试验证提供了极大的方便，用户不必再为构造信号产生函数、信号图形等方面费心思，可将主要精力放在算法的设计上。

MATLAB 信号处理箱覆盖了经典信号处理理论的大多数内容，是一个非常优秀的算法研究和辅助设计工具。它不仅提供了大量的工具函数，而且还提供交互式的滤波器分析、设计工具和谱分析工具。按照功能可以将信号处理工具箱中的函数分成以下 9 类：

1）信号和波形的产生；

2）FFT、DCT、希尔伯特和其他一些数学变换；

3）数字和模拟滤波器设计；

4）随机信号处理和经典谱估计；

5）参数模型功率谱估计；

6）多采样率信号处理；

7）窗函数；

8）时域-频域分析；

9）倒谱分析。

图像处理虽然从本质上是二维数字信号，但具有一些特殊的性质，尤其与人

的感知紧密联系在一起，MATLAB 提供了专门的工具箱，主要包括图像增强、除噪、分割和常用的变换等方面。

通信工具箱包含了通信系统设计、分析、仿真的工具函数和 Simulink 模块，是一个完整的理论研究和工程设计平台。其内容包括信号发生器、信源编码和量化、差错控制编码、调制和解调、基带和通带信道模型等。

Simulink 是 MATLAB 中的一种可视化仿真工具，是一种基于 MATLAB 的框图设计环境，是实现动态系统建模、仿真和分析的一个软件包，被广泛应用于线性系统、非线性系统、数字控制及数字信号处理的建模和仿真中。Simulink 可以用连续采样时间、离散采样时间或两种混合的采样时间进行建模，同时也支持多速率系统，也就是系统中的不同部分具有不同的采样速率。

Simulink 是用于动态系统和嵌入式系统的多领域仿真和基于模型的设计工具。对各种时变系统，包括通信、控制、信号处理、视频处理和图像处理系统，Simulink 提供了交互式图形化环境和可定制模块库来对其进行设计、仿真、执行和测试。

构架在 Simulink 基础之上的其他产品扩展了 Simulink 多领域建模功能，也提供了用于设计、执行、验证和确认任务的相应工具。Simulink 与 MATLAB 紧密集成，可以直接访问 MATLAB 大量的工具来进行算法研发、仿真的分析和可视化、批处理脚本的创建、建模环境的定制以及信号参数和测试数据的定义。

运用这些工具，可以方便地进行各种信号处理和现实，如下面这段代码实现了正弦函数的希尔伯特变换：

```
t=0:1/1023:1;                              % 产生 1024 个时间点
x=sin(2*pi*60*t);                          % 产生正弦取样波
y=hilbert(x);                              % 调用希尔伯特变换函数
plot(t(1:50),real(y(1:50)));               % 画出变换结果的实部
hold on                                    % 画图保持打开
plot(t(1:50),imag(y(1:50)),':');           % 画出变换结果的虚部
hold off                                   % 画图保持关闭
```

MATLAB 画出的图像如图 1-8 所示。

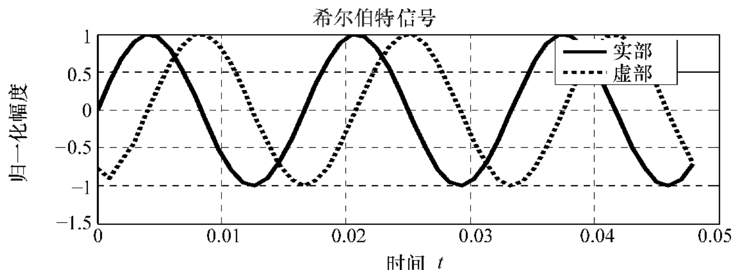


图 1-8　离散希尔伯特变换

下面这段代码则实现了 sinc 函数的傅里叶变换：

```
t=0:0.01:1;                                  % 产生 101 个采样时间点
x=sinc(2*pi*5*t);                            % 生成 sinc 函数
A=dftmtx(length(t));                         % 产生 DFT 变换矩阵
y=x*A;                                       % 对 x 进行 DFT 运算
subplot(1,2,1),plot(t,x,'LineWidth',2);      % 画出 sinc 信号
subplot(1,2,2),plot(t,y,'LineWidth',2);      % 画出频域图
```

MATLAB 画出的图像如图 1-9 所示。

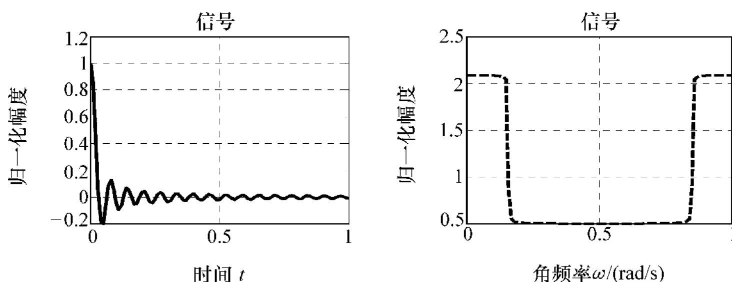


图 1-9　sinc 函数的 DFT 变换

1.4　本书的主要内容

本书包括三部分内容，第一部分是关于全相位数字滤波器设计及 MATLAB 实现，第二部分是关于全相位数字滤波器应用的挖掘及 MATLAB 实现，第三部分是关于全相位数字滤波器的基于 MATLAB Simulink 的仿真实验。

第一部分：用 MATLAB 实现无窗、单窗和双窗全相位数字滤波器及与其等效的其他三种系统；经过对全相位系统的详细分析和滤波实验后，给出 DCT 域、DWA 域中的实现形式；分析了窗在设计 FIR 滤波器中的重要作用，用 MATLAB 实现三种设计窗函数法（即基于 LMS、矩形二叉分解和最小均方误差准则）。

第二部分：用 MATLAB 实现全相位在自适应信号处理、二维模板设计、信号谱分析以及在多分辨率分析等方面的应用。首先，介绍了自适应滤波算法和滤波器结构，在分析了常规时域和频域 LMS 自适应算法后提出加权重叠的概念，继而提出时域和变换域加权重叠 LMS 算法，并应用于自适应除噪和辨识中；在二维 FOURIER 模板设计中结合重叠加权算法给出三种新型的模板结构，用来内插恢复亚奈图像，取得了良好的效果；谱分析和多分辨率分析中，均给出了加权重叠算法的理论分析并用 MATLAB 实验验证了结论的正确性。

第三部分：主要运用 MATLAB Simulink 子模块设计实现全相位数字滤波器进行了仿真验证。

1.5 本书的主要特点

本书共分成 10 章，其中前 5 章侧重全相位数字滤波器的架构实现和性质证明，这些实现和证明都通过 MATLAB 来进行验证；第 6 章 ~ 第 9 章侧重全相位滤波的应用，并与传统方法进行了对比，所有的应用实验都提供 MATLAB 代码；第 10 章借助 MATLAB Simulink 进行仿真，为 DSP 实现提供参考。

特点一：算法实现了无窗、单窗和双窗全相位数字滤波器，消除了数字信号处理中因分段带来的截断误差；提出并详尽地分析了系统的重要特性且给出理论证明和实验论证；给出了 APF 在离散余弦变换域、离散沃尔什变换域的结构形式，并进行了比较和分析。

特点二：提出了基于 LMS 准则的窗函数设计方案，它以有限抽样点处的均方误差作为评价系统的性能指标，使窗函数在 LMS 准则下收敛到最优。设计中提出了单位冲击响应加权的余弦基神经网络设计思想，该算法具有收敛速度快，易于控制的优点；提出了基于矩形二叉分解的窗函数设计方案，它将系统中的窗看成由较低阶的分别位于二叉树左右分支的矩形窗卷积而成，并利用回归定义形成多级分解，优化得到的窗是通过窗谱对滤波器过渡带宽度影响分析的结果。

特点三：提出了对信号做预处理加权重叠 W-O 算法。将 W-O 算法运用于时域和变换域自适应滤波器中，分别给出了基于 WO-LMS 和 WO-TLMS 准则的滤波算法并通过实验证明，带 W-O 预处理模块的自适应滤波系统不但加快了收敛速度，而且极大地降低了稳态均方失调。将 W-O 成功地运用于二维模板设计中，提出无窗、单窗和双窗 FOURIER 内插模板的概念，实验表明，带窗模板的性能优于无窗模板。将 W-O 算法成功运用于谱分析中，极大地减少了旁瓣的数量和幅度，提高了频谱分辨率和信号的检测能力。

特点四：本书在对全相位滤波理论总结的基础上，侧重其在窗函数设计、自适应信号处理、内插模板设计、谱分析和多分辨率分析等方面的运用，而抛开了繁冗的理论推导，全部采用 MATLAB 语言实现算法并提供测试仿真结果。

“重应用与实现”是本书的主要特点，书中提供的大量算法使结合本专业准备运用全相位滤波方法的读者具备了现成的分析和研究工具，也为正在研究全相位滤波理论的专家们提供了一定参考。

第 2 章 全相位理论基础

2.1　FIR 数字滤波器设计方法

数字滤波器(DF)的设计是数字信号处理的主要内容之一。数字滤波器与模拟滤波器一样，从频率响应分类可以划分为低通、高通、带通和带阻等滤波器，当然还有其他一些按照具体技术要求提出的滤波器。我们知道，序列的傅里叶变换具有以 2π 为周期的周期性，数字滤波器的特性也具有这一周期性，各种滤波器的幅度特性如图 2-1 所示。由图中可见，低通通带处于 0 或 2π 整数倍频率附近，高通通带处于 π 的奇数倍频率附近。

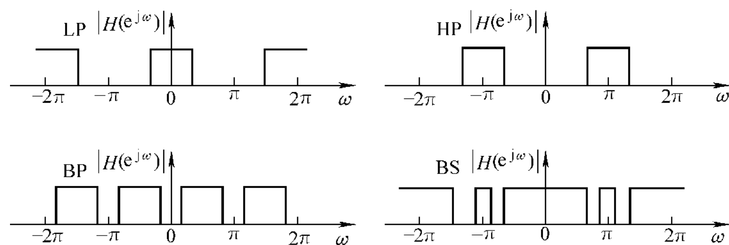


图 2-1　各种 DF 的幅度特性示意图

滤波器的设计任务就是根据预先规定的技术指标，找到系统传输特性 $\boldsymbol{H}$，使滤波器的性能满足技术指标。滤波器的技术指标可以在频域也可以在时域给出，但通常是在频域给出。图 2-1 所示的是一些理想非因果滤波器的幅频特性。实际设计中只能是尽量逼近它，只要满足指定的指标就可以了。一个实际滤波器的幅频特性在通带内允许有一定的波动，阻带衰减大于给定的衰减，且在通带、阻带之间允许有一定宽度的过渡带。图 2-2示出了一个低通的幅频特性。

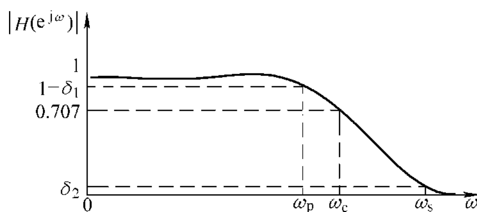


图 2-2　低通数字滤波器的幅频特性

通带范围是 $0 \leqslant \omega \leqslant \omega_p$，在通带内

要求 $1-\delta_1 \leqslant |H(e^{j\omega})| \leqslant 1$，$\omega_p$ 称为通带上限频率或滤波器的截止频率。频率范围 $\omega_s \leqslant \omega \leqslant \pi$ 称为阻带，在阻带内要求 $|H(e^{j\omega})| \leqslant \delta_2$，$\omega_s$ 称为阻带下限频率。频率在 ω_p 和 ω_s 之间称为过渡带，一般要求在过渡带幅频特性单调下降。通带衰减（波动）和阻带衰减（波动）通常用分贝（dB）表示。

$$
\begin{aligned}
\alpha_p &= 10\lg\left|\frac{H(e^{j0})}{H(e^{j\omega_p})}\right|^2 = -20\lg|H(e^{j\omega_p})| \\
\alpha_s &= 10\lg\left|\frac{H(e^{j0})}{H(e^{j\omega_s})}\right|^2 = -20\lg|H(e^{j\omega_s})|
\end{aligned}
\tag{2-1}
$$

式中，α_p 和 α_s 分别称为通带最大衰减和阻带最小衰减。如果 $\omega_c = 3\text{dB}$，则称为 3dB 截止频率，对应的带宽称为 3dB 带宽。

数字滤波器按单位脉冲响应长度可分成无限脉冲响应滤波器和有限脉冲响应滤波器两类，分别称为 IIR 滤波器和 FIR 滤波器；按照滤波器的实现方法可以分成递归滤波器和非递归滤波器两类。在实际的信号与信息处理中，如图像信号处理、数据传输等对线性相位要求较高，并从设计灵活度和控制性角度来考虑，本书主要研究 FIR 滤波器的设计与应用。

2.1.1 常规法

FIR 数字滤波器的输入和输出关系是

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k) \tag{2-2}$$

它所对应的频率响应为

$$H(e^{j\omega}) = H(z)\big|_{z=e^{j\omega}} = \sum_{n=0}^{N-1} h(n)e^{-jn\omega} = H_g(\omega)e^{j\varphi(\omega)} \tag{2-3}$$

式中，$H_g(\omega)$ 为 $H(e^{j\omega})$ 的增益，是一个可正可负的实函数；$\varphi(\omega)$ 为 $H(e^{j\omega})$ 的相频特性。具有线性相位系统的相位满足：

$$\varphi(\omega) = \beta - \alpha\omega \tag{2-4}$$

式中，α、β 为常数。具有线性相位的系统可保证在通频带内各种频率的信号都具有线性延迟，这样滤波后的信号就不会发生相位失真。这在诸如数据传输、语音综合等很多领域中都是非常希望得到的特性。

把式（2-3）两端用余弦函数展开有

$$
\begin{aligned}
H_g(\omega)\cos(\beta - \alpha\omega) &= \sum_{n=0}^{N-1} h(n)\cos(n\omega) \\
H_g(\omega)\sin(\beta - \alpha\omega) &= \sum_{n=0}^{N-1} h(n)\sin(n\omega)
\end{aligned}
\tag{2-5}
$$

把上面两式对应相除再交叉相乘展开，并利用三角恒等式得

$$\sum_{n=0}^{N-1} h(n)\{\sin[\beta - (\alpha - n)]\omega\} = 0 \tag{2-6}$$

式（2-6）恒成立的两组解为

$$\text{第一组解}\begin{cases}\beta = 0\\ \alpha = \dfrac{N-1}{2}\\ h(n) = h(N-1-n)\end{cases} \qquad \text{第二组解}\begin{cases}\beta = \pm\dfrac{\pi}{2}\\ \alpha = \dfrac{N-1}{2}\\ h(n) = -h(N-1-n)\end{cases} \tag{2-7}$$

当 N 分别取偶数和奇数时，FIR 数字滤波器的频率特性有四种不同的形式。

1） $h(n) = h(N-1-n)$，且 N 为奇数。

$$\begin{aligned}&\text{增益特性：}\quad H_g(\omega) = \sum_{n=0}^{(N-1)/2} a(n)\cos(n\omega)\\ &\text{相频特性：}\quad \varphi(\omega) = -\left(\frac{N-1}{2}\right)\omega\end{aligned} \tag{2-8}$$

式中，$a(n) = \begin{cases} h\left(\dfrac{N-1}{2}\right) & n=0\\ 2h\left(\dfrac{N-1}{2}-n\right) & n=1,2,\cdots,\dfrac{N-1}{2}\end{cases}$。

MATLAB 实现如下：

```
% unit impulse response sequence h(n)--- begin
h(1)=0.0625;
for k=2:(N+1)/2
  h(k)=h(k-1)*2;
  h(N+1-k)=h(k);
end
h((N+1)/2)=0.75;
h(N)=h(1);
% construct unit impulse response sequence h(n)---end

% weigth sequence a(n)---begin
a=[h((N+1)/2),2*h((N+1)/2-[1:(N-1)/2])];
% weigth sequence a(n)---end
w=0:pi/2^N:2*pi;% frequency sampling sequence w
nw=[0:(N-1)/2]'*w;% weigh the frequency n*w
cos w=cos(nw);% consine array
```

```
Hg =a* cosw;% get the amplitude frequency property Hg
wps =0:pi* 2/(N -1):pi;
pw = - (N -1)/2* wps;

% draw h(n),a(n),Hg andφ(w)---begin
subplot(1,2,1),stem(h,'fill','MarkerSize',6,'Color','k');
grid on,xlabel('a)h(n)');
subplot(1,2,2),stem(a,'fill','MarkerSize',6,'Color','k');
grid on,xlabel('b)a(n)');
figure,subplot(1,2,1),plot(w,Hg,'k','LineWidth',3),...
   set(gca,'Xtick',[0pi2* pi],'XTickLabel',{'0','pi','2pi'}),grid on;
xlabel('c)Hg(w)');
subplot(1,2,2),plot(wps,pw,'k','LineWidth',3),grid on;
xlabel('d)φ(w)');
% draw h(n),a(n),Hg and φ(w)---end
```

运行得到的结果如图 2-3 所示。

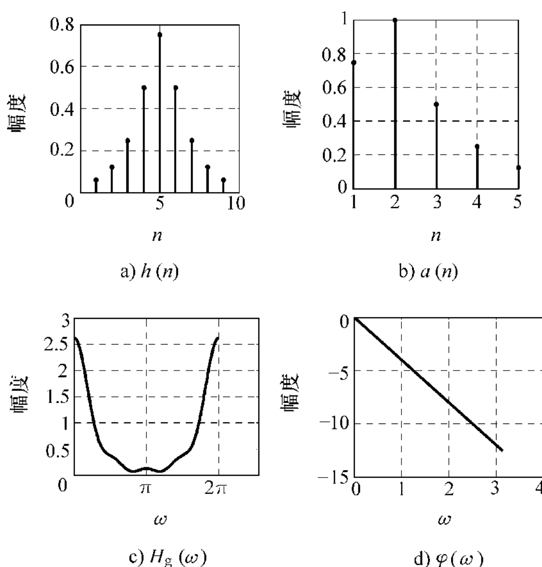


图 2-3 奇长度偶对称 $h(n)$ 设计 FIR 数字滤波器特性

2）$h(n)=h(N-1-n)$，且 N 为偶数。

$$
\begin{aligned}
&\text{增益特性：} H_g(\omega) = \sum_{n=1}^{N/2} b(n)\cos[(n-0.5)\omega] \\
&\text{相频特性：} \varphi(\omega) = -\left(\frac{N-1}{2}\right)\omega
\end{aligned}
\tag{2-9}
$$

式中，$b(n)=2h(N/2-n)\quad n=1,\ 2,\ \cdots,\ N/2$。

MATLAB 实现如下：

```
% unit impulse response sequence h(n)---begin
h(1)=0.0625;
for k=2:N/2
  h(k)=h(k-1)*2;
  h(N+1-k)=h(k);
end
h(N)=h(1);
% construct unit impulse response sequence h(n)---end

% weigth sequence b(n)---begin
b=2*h(N/2:-1:1);
% weigth sequence b(n)---end
w=0:pi/2^N:2*pi;% frequency sampling sequence w
nw=[0.5:N/2-0.5]'*w;% weigh the frequency n*w
cosw=cos(nw);% consine array
Hg=b*cosw;% get the amplitude frequency property Hg
wps=0:pi*2/(N-1):pi;
pw=-(N-1)/2*wps;
```

运行得到的结果如图 2-4 所示。

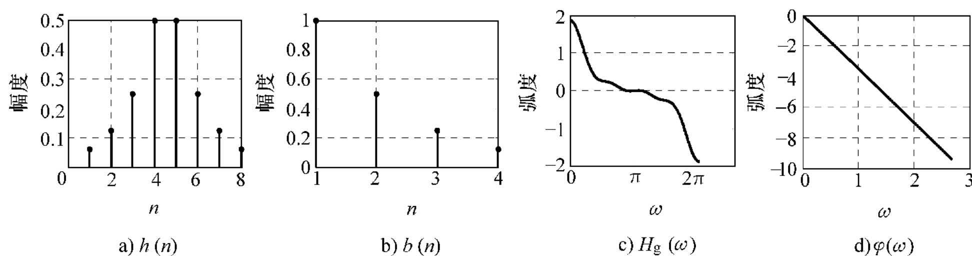


图 2-4　偶长度偶对称 $h(n)$ 设计 FIR 数字滤波器特性

3）$h(n)=-h(N-1-n)$，且 N 为奇数。

$$\text{增益特性：}\quad H_g(\omega)=\sum_{n=1}^{(N-1)/2}c(n)\sin(n\omega)$$
$$\text{相频特性：}\quad \varphi(\omega)=-\frac{\pi}{2}-\left(\frac{N-1}{2}\right)\omega \tag{2-10}$$

式中，$c(n)=2h\left(\frac{N-1}{2}-n\right)\quad n=1,\ 2,\ \cdots,\ \frac{N-1}{2}$。

MATLAB 实现如下：

```
% unit impulse response sequence h(n)---begin
h(1)=0.0625;
```

```
for k=2:N/2
  h(k)=h(k-1)*2;
  h(N+1-k)=-h(k);
end
h(N)=0;
% construct unit impulse response sequence h(n)---end

% weigth sequence c(n)---begin
c=2*h((N-1)/2:-1:1);
% weigth sequence c(n)--- end
w=0:pi/2^N:2*pi;% frequency sampling sequence w
nw=[1:(N-1)/2]'*w;% weigh the frequency n*w
sinw=sin(nw);% consine array
Hg=c*sinw;% get the amplitude frequency property Hg
wps=0:pi*2/(N-1):pi;
pw=-pi/2-(N-1)/2*wps;
```

运行得到的结果如图 2-5 所示。

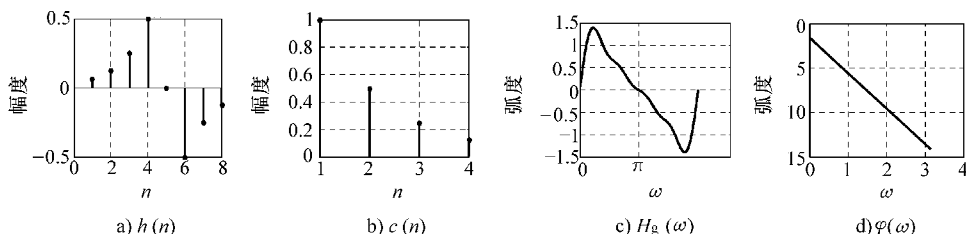


图 2-5 奇长度奇对称 $h(n)$ 设计 FIR 数字滤波器特性

4）$h(n)=-h(N-1-n)$，且 N 为偶数。

$$\text{增益特性：}\quad H_g(\omega)=\sum_{n=1}^{N/2} d(n)\sin[(n-0.5)\omega]$$
$$\text{相频特性：}\quad \varphi(\omega)=-\frac{\pi}{2}-\left(\frac{N-1}{2}\right)\omega \tag{2-11}$$

式中，$d(n)=2h\left(\frac{N}{2}-n\right)\quad n=1,\ 2,\ \cdots,\ \frac{N}{2}$。

MATLAB 实现如下：

```
% unit impulse response sequence h(n)---begin
h(1)=0.0625;
for k=1:N/2
  h(k+1)=h(k)*2;
  h(N+1-k)=-h(k);
end
```

```
% construct unit impulse response sequence h(n)---end

% weigth sequence d(n)---begin
d=2* h(N/2:-1:1);
% weigth sequence d(n)---end

w=0:pi/2^N:2* pi;% frequency sampling sequence w
nw=[0.5:N/2-0.5]'* w;% weigh the frequency n* w
sinw=sin(nw);% consine array
Hg=d* sinw;% get the amplitude frequency property Hg
wps=0:pi* 2/(N-1):pi;
pw=-pi/2-(N-1)/2* wps;
```

运行得到的结果如图 2-6 所示。

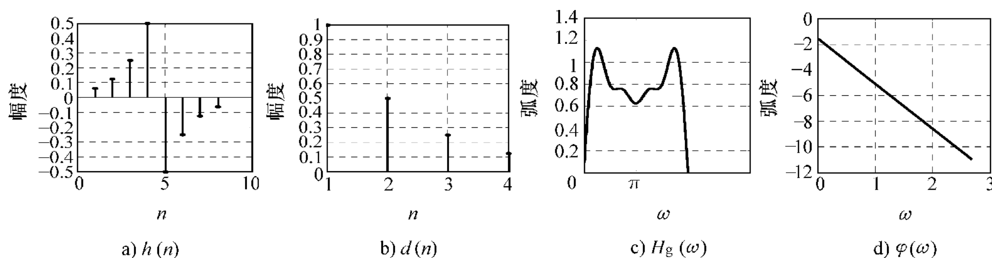


图 2-6　偶长度奇对称 $h(n)$ 设计 FIR 滤波器特性

综合以上 4 种条件对应的 FIR 滤波器设计，给出这种传统方法的 MATLAB 实现函数$[H_g,\varphi_\omega]=\mathrm{traFIR}(\boldsymbol{h},p)$。此函数包含两个输入参数即单位冲击响应 $\boldsymbol{h}$ 及滤波器类型 p，函数的输出为系统的幅频特性和相频特性。该函数的 MATLAB 实现如下：

```
% this function is to design the FIR by means of traditional method
% h is unit impulse response,p is the FIR type with value from 'a'to 'd'
% returned Hg is the amplitude to frequency,fw is the phase to frequency
function [Hg,fw]=traFIR(h,p)
clc,clf;
if nargin ~=2
  error('2 parameters should be input.');
  exit;
else
   errcode=chkpara(h,p);
   if errcode ~=0
      exit;
   end
```

```
end
N=length(h);
% frequency sampling sequence w and wps---begin
w=0:pi/2^N:2* pi;
wps=0:pi* 2/(N-1):pi;
% frequency sampling sequence w and wps---end
% weigth sequence wei(n),times frequency---begin
if(p=='a')||(p=='c')
    wei=[h((N+1)/2),2* h((N+1)/2-[1:(N-1)/2])];
    nw=[0:(N-1)/2]'* w;
else
    wei=2* h(N/2:-1:1);
    nw=[0.5:N/2-0.5]'* w;
end
% weigth sequence wei(n),times frequency---end
% sine or cosine array and phase-frequency propertyφ(w)---begin
if(p=='a')||(p=='b')
    sincosw=cos(nw);
    fw=-(N-1)/2* wps;
else
    sincosw=sin(nw);
    fw=-pi/2-(N-1)/2* wps;
end
% sine or cosine array and phase-frequency propertyφ(w)---end
Hg=wei* sincosw;% get the amplitude frequency property Hg
% draw h(n),wei(n),Hg andφ(w)---begin
subplot(1,2,1),stem(h,'fill','MarkerSize',6,'Color','k');
grid on,xlabel('a)h(n)');
subplot(1,2,2),stem(wei,'fill','MarkerSize',6,'Color','k');
grid on,xlabel(strcat('c)',p));
figure,subplot(1,2,1),plot(w,Hg,'k','LineWidth',3),...
    set(gca,'Xtick',[0 pi],'XTickLabel',{'0','pi'}),grid on;
xlabel('c)Hg(w)');
subplot(1,2,2),plot(wps,fw,'k','LineWidth',3),grid on;
xlabel('d)φ(w)');
% draw h(n),d(n),Hg andφ(w)---end
```

其中函数 chkpara 用于检测输入参数的合法性，这里不再赘述。

传统方法设计 FIR 滤波器需要知道系统的单位冲击响应序列，需要变换到频域才可以了解滤波器是低通或高通，这给设计带来了不方便。此外，此方法得到的滤波器在通带、阻带存在幅度较大的波动，总体性能比较低下。

2.1.2　窗函数法

带限信号的单位冲击响应 $h(n)$ 的持续时间是 $-\infty \sim \infty$，在物理上是无法实现的。对 $h(n)$ 截断为有限 $2M+1$ 项后右移 M 项，可变成因果性序列，即

$$h_{\mathrm{M}}(n)=h(n-M) \quad n=0,\ 1,\ \cdots,\ 2M \tag{2-12}$$

用 $h_{\mathrm{M}}(n)$ 近似代替 $h(n)$ 研究系统特性存在误差，记逼近误差为 ε^2，则

$$\varepsilon^2 = \frac{1}{2\pi}\int_{-\pi}^{\pi} |E(\mathrm{e}^{\mathrm{j}\omega})|^2 \mathrm{d}\omega \tag{2-13}$$

式中，$\boldsymbol{E}$ 为 $\boldsymbol{h}_{\mathrm{M}}$ 的幅频特性 $|\boldsymbol{H}_{\mathrm{M}}|$ 和 $\boldsymbol{h}$ 的幅频特性 $|\boldsymbol{H}|$ 差值。把 $\boldsymbol{H}_{\mathrm{M}}$ 和 $\boldsymbol{H}$ 展开成傅里叶级数有

$$\begin{aligned}
H_{\mathrm{M}}(\mathrm{e}^{\mathrm{j}\omega}) &= \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n\cos(n\omega) + \sum_{n=1}^{\infty} b_n\sin(n\omega) \\
H(\mathrm{e}^{\mathrm{j}\omega}) &= \frac{A_0}{2} + \sum_{n=1}^{M} A_n\cos(n\omega) + \sum_{n=1}^{M} B_n\sin(n\omega)
\end{aligned} \tag{2-14}$$

把式（2-14）代入到式（2-13）中并利用三角函数的正交性得到

$$\varepsilon^2 = \frac{(a_0-A_0)^2}{2} + \sum_{n=1}^{M}(a_n-A_n)^2 + \sum_{n=1}^{M}(b_n-B_n)^2 + \sum_{n=M+1}^{\infty}(a_n^2+b_n^2) \tag{2-15}$$

只有当 $A_0=a_0$，$A_n=a_n$，$B_n=b_n$，$n=1,\ 2,\ \cdots,\ M$ 时 ε^2 才最小。亦即用 $\boldsymbol{H}_{\mathbf{M}}$ 来近似 $\boldsymbol{H}$ 的最小误差的条件是 $\boldsymbol{H}_{\mathbf{M}}$ 的单位抽样响应必须是 $\boldsymbol{H}$ 的傅里叶系数。所以，有限项傅里叶系数是在最小二次方意义上对原信号的最佳逼近，且 M 取得越大，误差 ε^2 越小。

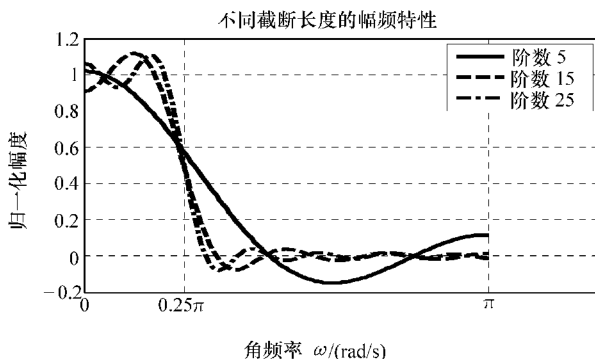


图 2-7　3 种截断长度对应的滤波器特性

图 2-7 是 M 分别取 5、10 及 20 时得到的一截止频率为 0.25π 低通滤波器的幅频特性曲线。当 M 取不同值时，$\boldsymbol{H}_{\mathbf{M}}$ 都在不同程度上近似于 $\boldsymbol{H}$。M 过小时，通频带过窄且阻带内波纹较大。M 增大时，通频带接近 0.25π 且阻带内波纹减小，但在通带内出现了波纹，随着 M 的继续增大，这些波纹并不能消失，只是最大的上冲越来越接近间断点，这就是吉布斯现象。

MATLAB 实现如下：

```
% This function draw FIR with different rank N1/N2/N3
% Using sinx/x to get h with limited length
```

```
% to testify the FIR property to rank

function winFIR(N1in,N2in,N3in,WPin)
clc;
if nargin = =0
  N1 =5;N2 =15;N3 =25;
  wp =0.25* pi;
else if narin ~ =4
      error('4 arguments should be provided.');
      exit;
   else
      N1 =N1in;N2 =N2in;N3 =N3in;
      wp =WPin;
   end
end
ws =0:pi/128:pi;
Hg =makeHg(ws,[N1,N2,N3],wp);

lgndstr =num2str([N1;N2;N3]);
figure,plot(ws,Hg,'LineWidth',2,'Color','k'),
legend(strcat('½×Êy',lgndstr));
xlabel('\omega(rad/s)'),ylabel('normalized amp');
wps =strcat(num2str(wp/pi),'pi');
set(gca,'Xtick',[0 wp pi],'XTickLabel',{'0',wps,'pi'});

function Hg =makeHg(w,N,wp)
T =length(N);
if T = =0
   warning('Bad parameter passing:N');
   return;
end
for t =1:T
   n =[0:N(t) -1] - (N(t) -1)/2;
   h =wp/pi* sinc(wp/pi* n);
   a =2* h((N(t) +1)/2:end);
   a(1) =h((N(t) +1)/2);
   wn = (w.')* [0:(N(t) -1)/2];
   cosw =cos(wn);
   res =cosw* diag(a);
   Hg(t,:) =sum(res,2);
end
```

对 $\boldsymbol{h}_M$ 的自然截断等于对它施加了一个矩形窗口 $\boldsymbol{w}$，在频域则相当于 $\boldsymbol{H}_M$ 与 $\boldsymbol{w}$ 的频谱 $\boldsymbol{W}$ 的线性卷积。矩形窗函数的频谱有较大的旁瓣，正是这些旁瓣在卷积时产生了吉布斯现象。为了减小吉布斯现象，可以选取一些旁瓣较小的窗口，如三角窗、汉宁窗等来代替矩形窗。图 2-8 中给出了 $M=10$ 时分别加矩形窗和汉宁窗后的幅频特性曲线。

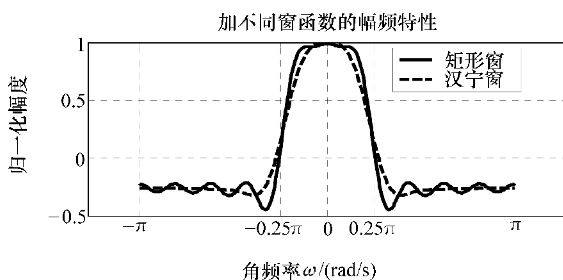


图 2-8　加矩形窗和汉宁窗后的幅频特性

MATLAB 实现如下：

```
% This function draw FIR with different rank N and different window W
% Usingsinx/x to get h with limited length to display the FIR property
% improvement aftering being windowed
% rectangle,triangle,consie,hanning,hamming,blackman,papoulis

function winComp(Nin,Win)
clc;clf;
WT = char('rectangle','triangle','bartlett','hanning','hamming','blackman');
Windex = ['r','t','b','h','m','c'];
if nargin = =0
  N =20;
  W = find(Windex = ='h');
  Wts =0.25* pi;
else if narin ~ =2
      error('2 arguments should be provided.');
      exit;
   else
      N = Nin;
      W = Win;
   end
end
WIN = [boxcar(N)triang(N)bartlett(N)hanning(N)hamming(N)...
     blackman(N)];
WIN = WIN';
ws = -pi:pi/128:pi;
Hg = makeHg(ws,N,Wts,WIN([1W],:));
plot(ws,Hg,'LineWidth',3,'Color','k'),legend('¾øDÎ'°',WT(W,:)),grid on;
xlabel('\omega(rad/s)'),ylabel('normalized amp');
```

```
wps = strcat(num2str(Wts/pi),'pi');
set(gca,'Xtick',[-pi -Wts0Wtspi],'XTickLabel',{'-pi',strcat('-',wps),'0',
wps,'pi'});
function Hg = makeHg(w,N,Wts,WIN)
T = size(WIN,1);
if T = =0
  warning('Bad parameter passing:N');
  return;
end
for t =1:T
    n = [0:N -1] - (N -1)/2;
    h =Wts/pi* sinc(Wts/pi* n);
    hwined =h. * WIN(t,:);
    if mod(N,2) = =  0
        a =2* hwined(N/2 +1:end);
        wn = (w. ')* [1:N/2];
    else
        a =2* hwined((N +1)/2:end);
        a(1) =hwined((N +1)/2);
        wn = (w. ')* [0:(N -1)/2];
    end
    cosw = cos(wn);
    res =cosw* diag(a);
    Hg(t,:) =sum(res,2);
end
```

在 MATLAB 中窗函数有矩形窗、三角窗、巴特利特窗、汉明窗、汉宁窗、布莱克曼窗、切比雪夫窗和凯泽窗，对应的函数波形和频谱曲线如图 2-9 所示。

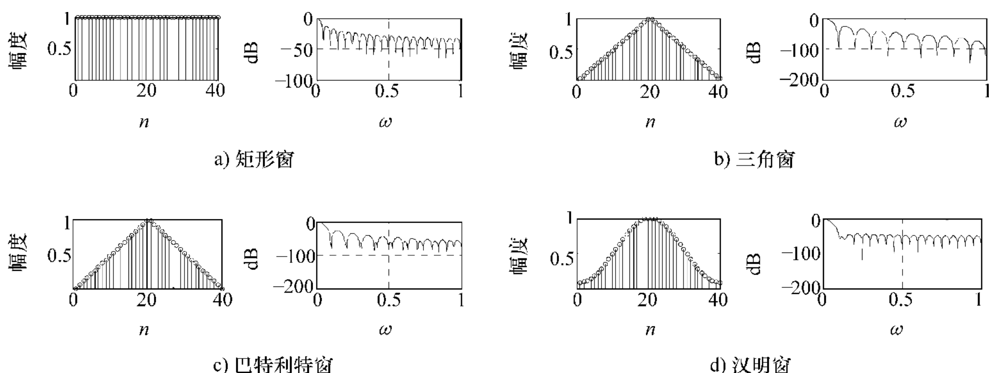


图 2-9　8 种窗函数的波形和频谱图对比

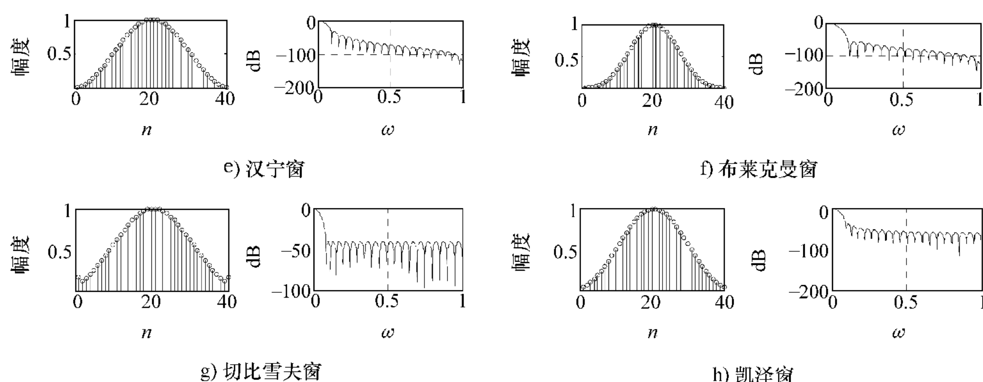


图 2-9　8 种窗函数的波形和频谱图对比（续）

8 种窗函数的 MATLAB 实现如下：

```
% to draw rectangle,triangular,bartlett,hamming,hanning,blackman,
% chebwin and kaier window and spectrum
function wn =winfun(N)
if nargin = =0
    N =40;
end
wins = [boxcar(N)triang(N)bartlett(N)hamming(N)hanning(N)...
    blackman(N)chebwin(N,40)kaiser(N,5)];
winame =char('a.rectangle','b.triangle','c.bartlett','d.hamming',...
    'e.hanning','f.blackman','g.chebwin','h.kaiser');
nwin =size(wins,2);
ncol =4;nrow = (nwin +mod(nwin,2))/2;
for n =1:nwin
    [h,w] =freqz(wins(:,n),1);
    subplot(nrow,ncol,2* n -1),stem(wins(:,n),'k','MarkerSize',3);
    title(winame(n,:));
    subplot(nrow,ncol,2* n),plot(w/pi,20* log10(abs(h)/abs(h(1))),'k');
    title(winame(n,:)),grid on;
end
```

通过窗函数使滤波器的波纹数得到了减少，但却增加了过渡带宽度。另外，设计滤波器的任务又增加了窗函数的设计。窗函数法是针对时域幅度的控制，对频域期望的特性不能得到有效控制。

2.1.3　频率抽样法

设要设计的 FIR 数字滤波器的频率响应是 $H(e^{j\omega})$，对其进行采样使每一个

周期有 N 个抽样值，即

$$H(k)=H(\mathrm{e}^{\mathrm{j}\omega_k})\Big|_{\omega_k=\frac{2\pi}{N}k}=H(\mathrm{e}^{\mathrm{j}2k\pi/N})=H_{\mathrm{g}}(k)\mathrm{e}^{\mathrm{j}\varphi_{\mathrm{g}}(k)} \tag{2-16}$$

式中，$\boldsymbol{H}_{\mathrm{g}}(k)$是滤波器的增益；$\varphi_{\mathrm{g}}(k)$是其相位响应。对 $\boldsymbol{H}$ 作 IDFT 得到单位抽样响应序列为

$$h(n)=\frac{1}{N}\sum_{k=0}^{N-1}H(k)\mathrm{e}^{\mathrm{j}\frac{2\pi}{N}kn}\quad n=0,1,\cdots,N-1 \tag{2-17}$$

由此可构造滤波器的转移函数 $H(z)$。为保证 $\boldsymbol{h}$ 是实数及 $\boldsymbol{H}$ 具有线性相位则要求

$$H_{\mathrm{g}}(k)=\begin{cases}H_{\mathrm{g}}(N-k) & N\text{ 为奇数}\\ -H_{\mathrm{g}}(N-k) & N\text{ 为偶数}\end{cases}\quad k=0,1,\cdots,M$$

$$M=\begin{cases}(N-1)/2 & N\text{ 为奇数}\\ N/2 & N\text{ 为偶数}\end{cases} \tag{2-18}$$

$$\varphi_{\mathrm{g}}(k)=-(N-1)k\pi/N\quad k=0,1,\cdots,M \tag{2-19}$$

设 16 阶低通滤波器的频率采样幅值序列 $\boldsymbol{H}_{\mathbf{g}}=[1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1]$，由 MALAB 可得到单位冲击响应序列 $\boldsymbol{h}$、幅频特性 $H_{\mathrm{g}}(\omega)$和相频特性 $\varphi_{\mathrm{g}}(\omega)$，如图 2-10 所示。

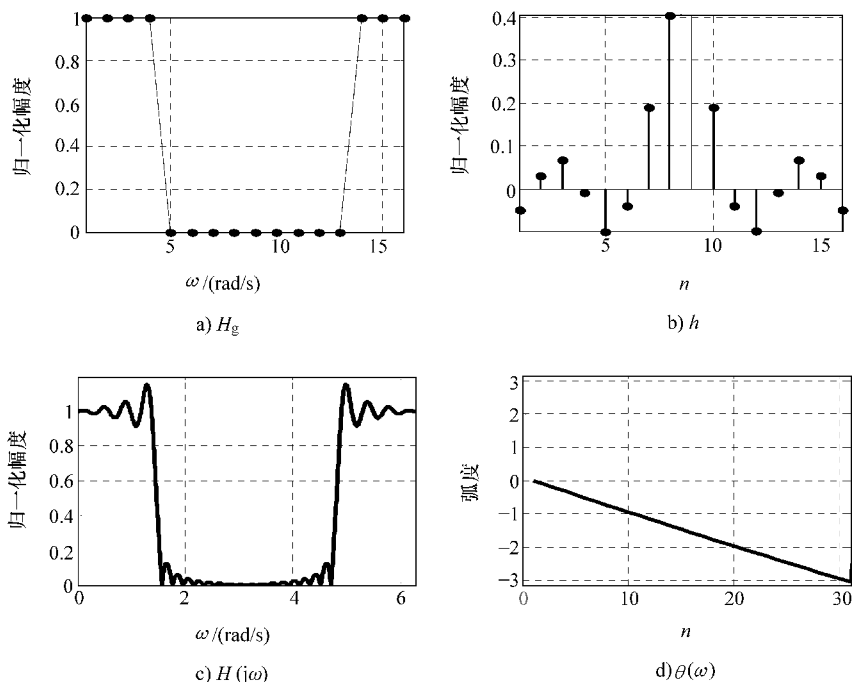


图 2-10 频率采样法得到的滤波器

MATLAB 实现代码如下：

```
% design FIR filter by means of frequency sampling
function h = freqFIR(N)
if nargin = =0
   N =16;
End

wps = pi/2;
hones = floor(wps/(2* pi/N));
Hg = zeros(1,N);Hg(1) =1;Hg(2:hones) =1;
Hg(N: -1:N +2 - hones) = ( -1)^(mod(N,2));
Pg(1:N) = - (N -1)/N* pi* [0:(N -1)];
H = Hg. * exp(j* Pg);%  system frequency-phase property
h = real(ifft(H,N));%  system unit impulse response
```

频率采样法是从频域进行设计，物理概念清楚。但频率采样点都局限在 $2\pi/N$ 的整数倍点上，在指定某特殊截止频率点时就要增大阶数 N 以充分接近频率点，计算量和复杂性都会增加。

2.1.4　切比雪夫逼近法

如图 2-11 所示，设所希望的理想频率响应为

$$H_d(e^{j\omega}) = \begin{cases} 1 & 0 \leqslant \omega \leqslant \omega_p \\ 0 & \omega_s \leqslant \omega \leqslant \pi \end{cases} \tag{2-20}$$

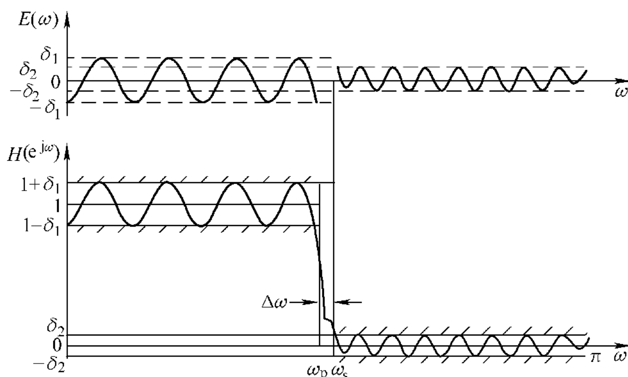


图 2-11　低通数字滤波器的一致逼近

δ_1、δ_2 分别为通带和阻带波纹峰值。由 Chebyshev 交错点定理可知，如果 $H(e^{j\omega})$ 是对 $H_d(e^{j\omega})$ 的最佳一致逼近，则 $H(e^{j\omega})$ 应具有如图 2-11 的等波纹性质。定义加权函数为

$$W(\omega)=\begin{cases}1/k & 0\leqslant\omega\leqslant\omega_{\mathrm{p}},\ k=\delta_1/\delta_2\\ 1 & \omega_{\mathrm{s}}\leqslant\omega\leqslant\pi\end{cases} \tag{2-21}$$

对于满足线性相位的奇数阶系统，逼近误差为

$$E(\omega)=W(\omega)\left|\sum_{n=0}^{M}a(n)\cos(n\omega)-H_{\mathrm{d}}(\mathrm{e}^{\mathrm{j}\omega})\right| \tag{2-22}$$

若已知 $M+2$ 个交错频率 ω_0、ω_1、…、ω_{M+1}，则满足：

$$W(\omega_k)\left[H_{\mathrm{d}}(\omega_k)-\sum_{n=0}^{M}a(n)\cos(n\omega_k)\right]=-(-1)^k\rho\quad k=0,1,\cdots,M+1 \tag{2-23}$$

式中，$\rho=E_n=\max|E(\omega)|$。从而构建 $M+2$ 个方程求解出 a_0、a_1、…、a_M 和 ρ，这样最佳滤波器就可构成。在 MATLAB 中利用 Remez 算法求解方程组，用一次次的迭代来求得一组交错点组，在整个运算过程中占用内存较少且运算时间较短。

切比雪夫逼近法的效率依赖于初始极值频率点的估计，通带和阻带内波纹数较多，这是其两个主要缺点。

2.1.5 其他方法

遗传算法是在 20 世纪 70 年代初期由美国 Michigan 大学的 Holland 教授发展起来的，它是一种模仿生物进化过程的结构型随机搜索。主要思想就是通过选择合适的适应度函数，采用遗传算法的选择、交叉和变异准则进行不断地筛选，直到结果达到规定的指标为止。其优点是得到的解是全局的，是在所选适应度函数下的最优解；缺点是由于计算中要反复变异实验，计算量较大，而且适应度函数的数学模型建立不同，结果也将不同，甚至差别很大，如何确定合适的适应度函数是遗传算法的关键。

神经网络是模仿和延伸人脑认知功能的新型智能信息处理系统，它是由大量的简单处理单元（即神经元）所构成的一个巨复杂的非线性自适应动力学习系统。基于神经网络的 FIR 数字滤波器设计的主要思想是使神经网络输出的幅频特性与理想幅频特性之间的全局误差在通带和阻带范围内最小。神经网络是按照 FIR 线性相位滤波器的四种形式，以加权余弦作为神经元，以频率采样点作为输入，相应的幅频响应作为输出来建立的。其优点是在通带内没有过冲和波动现象，可得到大的阻带衰减；缺点是阻带中波纹数较多，过渡带较宽。神经网络的解也是全局的，但目标函数的选取以及解的唯一性是神经网络算法的存在的两个理论问题。

2.2 全相位数字滤波器

$x(n)$ 的 DFT 系数 $X(k)$ 是频域中各次谐波频率处的谱线即 $X(0)$ 为直流谱线，

$X(1)$为基频谱线，$X(2)$为二次谐波谱线，…，所以 DFT 可以看做是许多窄带带通滤波器组成的滤波器。将 $x(n)$的 DFT 展开如下：

$$\begin{aligned}X(k) &= \sum_{n=0}^{N-1} x(n)\mathrm{e}^{-\mathrm{j}\frac{2\pi}{N}kn} \\ &= x(0) + x(1)\mathrm{e}^{-\mathrm{j}\frac{2\pi}{N}k} + x(2)\mathrm{e}^{-\mathrm{j}\frac{2\pi}{N}2k} + \cdots + x(N-1)\mathrm{e}^{-\mathrm{j}\frac{2\pi}{N}(N-1)k}\end{aligned} \tag{2-24}$$

$X(k)$可以看做是 $x(n)$与一个滤波器的脉冲响应 $h_k(n)$的卷积，即

$$X(k) = \sum_{n=0}^{N-1} x(n)h_k(N-1-n) \tag{2-25}$$

此滤波器的脉冲响应为

$$h_k(n) = \begin{cases} \mathrm{e}^{-\mathrm{j}\frac{2\pi}{N}(N-1-n)} & n=0,1,2,\cdots,N-1 \\ 0 & n<0 \text{ 和 } n \geqslant N \end{cases} \tag{2-26}$$

由式（2-24）可知，滤波器在 $N-1$ 个时刻的输出是 $X(k)$，如图 2-12 所示。

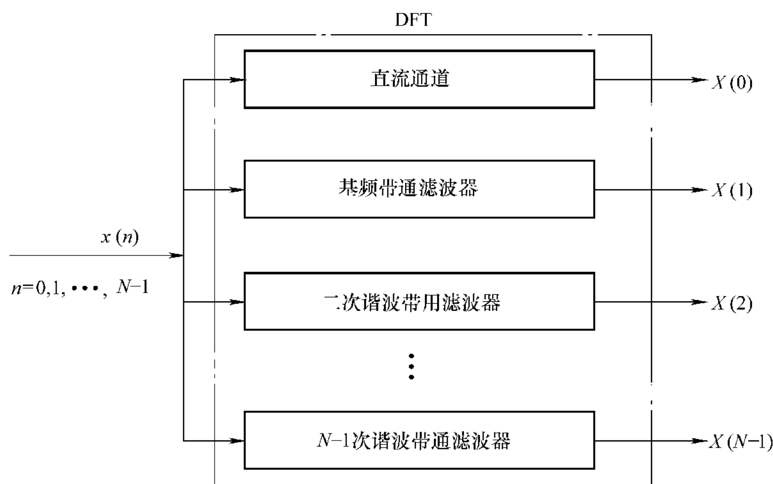


图 2-12　DFT 等效为 N 个带通滤波器

对滤波器的脉冲响应做 Z 变换可得

$$\begin{aligned}H_k(z) &= Z(h_k(n)) = \sum_{n=0}^{N-1} h_k(n)z^{-n} = \sum_{n=0}^{N-1} \mathrm{e}^{-\mathrm{j}\frac{2\pi}{N}(N-1-n)} z^{-n} \\ &= \mathrm{e}^{-\mathrm{j}\frac{2\pi}{N}(N-1)} \sum_{n=0}^{N-1} \left(\mathrm{e}^{-\mathrm{j}\frac{2\pi}{N}k} z\right)^{-n} = \mathrm{e}^{-\mathrm{j}\frac{2\pi}{N}(N-1)} \frac{1 - z^{-N}\left(\mathrm{e}^{\mathrm{j}\frac{2\pi}{N}k}\right)^N}{1 - z^{-1}\left(\mathrm{e}^{\mathrm{j}\frac{2\pi}{N}k}\right)}\end{aligned} \tag{2-27}$$

令 $z=\mathrm{e}^{\mathrm{j}\Omega T}=\mathrm{e}^{\mathrm{j}2\pi f/f_s}$，得到滤波器的幅频特性如下：

$$\left|H_k\left(Z=\mathrm{e}^{\mathrm{j}2\pi\frac{f}{f_s}}\right)\right|=\left|\frac{1-\left(\mathrm{e}^{\mathrm{j}2\pi\frac{f}{f_s}}\right)^{-N}\left(\mathrm{e}^{\mathrm{j}\frac{2\pi}{N}k}\right)^{N}}{1-\left(\mathrm{e}^{\mathrm{j}2\pi\frac{f}{f_s}}\right)^{-1}\left(\mathrm{e}^{\mathrm{j}\frac{2\pi}{N}k}\right)}\right|=\left|\frac{\sin\left[\pi N\left(\frac{f}{f_s}-\frac{k}{N}\right)\right]}{\sin\pi\left(\frac{f}{f_s}-\frac{k}{N}\right)}\right| \tag{2-28}$$

当 $N=4$ 时 DFT 滤波器的特性曲线如图 2-13 所示。离散傅里叶变换相当于用中心频率为 kf_s/N，频率响应形式为 $\sin(N\cdot)/\sin(\cdot)$ 的 N 个滤波器对输入序列进行滤波。如果输入序列 $f(n)$ 中含有不是 $2\pi f_s/N$ 整数倍的频率分量，则经过傅里叶变换后就有可能从各个滤波器的输出端看到它的输出响应。为了抑制这种“频谱泄漏”，可以通过重叠滤波的方式对信号进行处理。

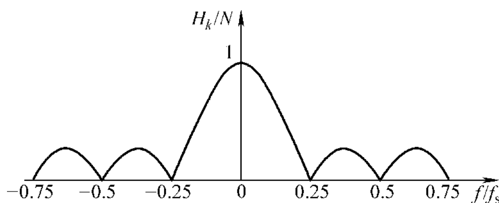


图 2-13　$N=4$ 时 DFT 滤波器的幅频特性

2.2.1　DFT 域无窗全相位数字滤波器

传统的 N 阶 FIR 滤波器某一时刻的输出由该时刻的输入以及它以前的 $N-1$ 个输入所决定。设系统单位冲击响应为 $\boldsymbol{h}_n=[h(0)h(1)\cdots h(N-1)]^{\mathrm{T}}$，$n$ 时刻输入为 $\boldsymbol{x}_n=[x(n)x(n-1)\cdots x(n-N+1)]^{\mathrm{T}}$，$n$ 时刻的块输出中对应 $x(n)$ 的输出等于 $\boldsymbol{x}_n$ 与 $\boldsymbol{h}$ 的循环卷积中对应 n 时刻的输出，其频域表示如图 2-14所示。$\boldsymbol{X}$ 和 $\boldsymbol{H}$ 分别是输入和单位响应的 N 点离散傅里叶变换。非重叠的滤波输入分段进行，且 $y(n)$ 只是对应输入 $x(n)$ 频率为 $2\pi(N-1)/N$ 的输出，因此无法求出系统的整体传输特性且输出段与段拼接处会出现不连续的边界效应。

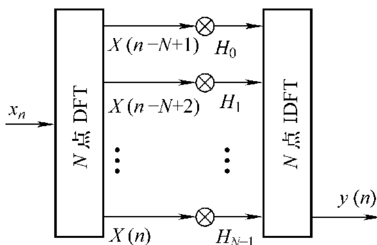


图 2-14　非重叠频域滤波

全相位滤波器是一个 N 阶的重叠滤波器，n 时刻的输出 $y(n)$ 是 N 个相关输入 $\boldsymbol{x}_{n+i}(i=0,1,\cdots,N-1)$ 块分别对应 $x(n)$ 的输出响应和的算术平均。设 $\boldsymbol{X}=[\boldsymbol{x}_n\quad \boldsymbol{x}_{n+1}\cdots\boldsymbol{x}_{n+N-1}]$ 是与输入 $x(n)$ 相关的矩阵，$\boldsymbol{Y}=[\boldsymbol{y}^{N-1}(n)\boldsymbol{y}^{N-2}(n)\cdots\boldsymbol{y}^{0}(n)]^{\mathrm{T}}$是 $x(n)$ 遍历角频率 $\omega=2k\pi/N(k=0，1，\cdots，N-1)$ 时 N 个输出。N 阶傅里叶频域全相位如图 2-15 所示。

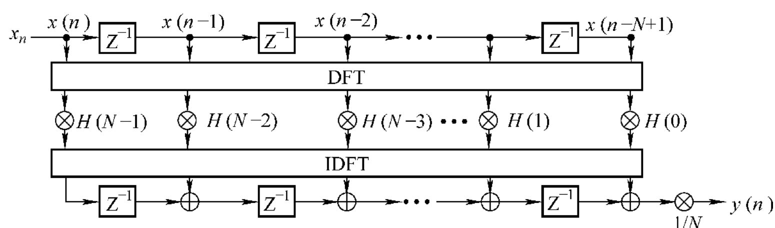


图 2-15　N 阶频域无窗全相位滤波器(ap0)

N 阶重叠滤波器在 n 时刻的输出 $y(n)$ 是由 $x(n)$ 以及它前后各 $N-1$ 个输入决定的，数据关系可表示如下：

$$
\begin{array}{cccccccc}
\boldsymbol{x}_n & \rightarrow & x(n) & x(n-1) & \cdots & x(n-N+1) & \rightarrow & y^{(0)}(n) \\
\boldsymbol{x}_{n+1} & \rightarrow & x(n+1) & x(n) & \cdots & x(n-N+2) & \rightarrow & y^{(1)}(n) \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\boldsymbol{x}_{n+N-1} & \rightarrow & x(n+N-1) & x(n+N-2) & \cdots & x(n) & \rightarrow & y^{(N-1)}(n)
\end{array}
$$

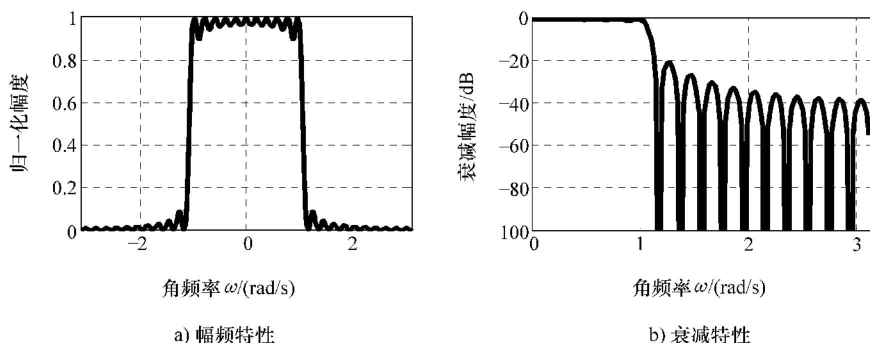
设输入信号为频率从 $-\pi$ 到 π 的余弦信号，每次取样 256 个点，32 阶系统特性 $\boldsymbol{H}=[1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1]$。

MATLAB 实现代码如下：

```
% this function performs to draw out the amplitude frequency
% of the frequency domain filter with consin input...
function out =apDFT0(N)
if nargin = =0
  N =32;samlen =N* 8;
end
H =[1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1];
w = -pi:2* pi /samlen:pi;t = -N +1:samlen;
s =cos(w'* t);Hs =repmat(H,size(w',1),1);
d =zeros(size(w',1),N);
for i =1 :samlen -N +1
    infft =fft(s(:,i:i +N -1),[],2);% input fft
    outfft =Hs. * infft;%  filtering the input
    out =real(ifft(outfft,[],2));%  input ifft
    if(i >N -1)
        y(:,i) =abs(d(:,1) +out(:,1))/N;%  final output
    end
    for j =1:N -2
        d(:,j) =d(:,j +1) +out(:,j +1);%  other middle outputs
    end
    d(:,N -1) =out(:,N);%  the most left out
```

```
end
yy=max(y,[],2);subplot(1,2,1),h1=plot(w,yy,'k','LineWidth',2);
grid on,axis([-pi pi 0 1.01]);
subplot(1,2,2),h2=plot(w,20*log10(yy),'k','LineWidth',2);grid on,axis([0
pi -100 0]);
if nargout==1
   out=[w,yy];
end
```

对不同频率的余弦信号经过全相位滤波器处理而产生的输出表明了系统的传统特性，上述算法运行结果如图 2-16 所示。



a) 幅频特性　　b) 衰减特性

图 2-16　32 阶无窗全相位滤波器特性

全相位系统对频率为 0. 9817rad/s 左右信号进行了选择，通过 $\boldsymbol{H}$ 的设置可看出，上截止频率也正是 $2\pi/32\times5\approx0.9817$rad/s，从而验证了设定为 $\boldsymbol{H}$ 的全相位系统具有低通特性。现定义抽取算子 $\boldsymbol{S}=[s_0 s_1\cdots s_{N-1}]^{\mathrm{T}}$，其中 s_i表示第 i 个元素为 1 其余元素为 0 的列向量。系统频域取样矩阵 $\boldsymbol{H}=\mathrm{diag}(H_0 H_1\cdots H_{N-1})$，则图 2-15中重叠滤波器的输出为

$$y(n)=\frac{1}{N^2}\sum_{i=0}^{N-1}\left[\boldsymbol{s}_{N-1-i}^{\mathrm{T}}\cdot\boldsymbol{W}_N^*\cdot\boldsymbol{H}\cdot\boldsymbol{W}_N\cdot\boldsymbol{x}_{n+i}\right]\tag{2-29}$$

式中，$\boldsymbol{W}_N$ 是 N 阶傅里叶变换矩阵；$\boldsymbol{W}_N^*$ 是 $\boldsymbol{W}_N$ 的共扼。

将式（2-29）展开得

$$y(n)=\frac{1}{N}\sum_{i=0}^{N-1}\left\langle\sum_{j=0}^{N-1}\left\{\left[\frac{1}{N}\sum_{k=0}^{N-1}(w_N^{k(i+1+j)}H_k)\right]x_{n+i}(j)\right\}\right\rangle$$

式中，$w_N=\mathrm{e}^{-\mathrm{j}\frac{2\pi}{N}}$。注意到：

$$\begin{cases}\dfrac{1}{N}\displaystyle\sum_{k=0}^{N-1}w_N^{k(i+1+j)}H_k=\dfrac{1}{N}\displaystyle\sum_{k=0}^{N-1}w_N^{-k(-i-1-j)}H_k=h(-i-1-j)\\ \boldsymbol{x}_{n+i}(j)=x(n+i-N+1+j)\end{cases}\tag{2-30}$$

将式（2-30）代入到式（2-29）（注意 $h(n)$ 隐含具有周期 N）得

$$
\begin{aligned}
y(n) &= \frac{1}{N}\sum_{i=0}^{N-1}\left\{\sum_{j=0}^{N-1}[h(-i-1-j)x(n+i-N+1+j)]\right\} \\
&= \frac{1}{N}\sum_{i=0}^{N-1}\left\{\sum_{k=-i}^{-i+N-1}[h(k)x(n-k)]\right\},\ \text{其中}\ k=-i+N-1-j \\
&= \frac{1}{N}\left(\sum_{k=0}^{N-1}+\sum_{k=-1}^{N-2}+\cdots+\sum_{k=-N+1}^{0}\right)[h(k)x(n-k)] \qquad (2\text{-}31) \\
&= \frac{1}{N}\sum_{k=-N+1}^{N-1}[(N-|k|)h(k)x(n-k)] \\
&= \sum_{k=-N+1}^{N-1}[h(k)\bar{x}(n-k)],\ \text{其中}\ \bar{x}(n-k)=\frac{N-|k|}{N}x(n-k)
\end{aligned}
$$

即 N 阶频域全相位滤波器与 $2N-1$ 阶加三角窗的 FIR 滤波器等效，此 FIR 滤波器的单位冲击响应是全相位滤波器特性 $\boldsymbol{H}$ 的反离散傅里叶变换的两个周期延拓，第一等效图如图 2-17 所示。它的物理意义是：信号输入序列 $\boldsymbol{X}$ 中与 $x(n)$ 有关的 N 个块 $\boldsymbol{x}_{n+i}$，依次作为冲击响应分别为将 $\boldsymbol{h}$ 循环右移 i 位的 FIR 滤波器的输入，相对应的 N 个输出之和。

图 2-17　N 阶无窗全相位滤波器第一等效图（ap1）

以下 MATLAB 代码验证了图 2-17 与图 2-15 的 DFT 域全相位等效。输入信号由 $f_1=16\text{Hz}$ 和 $f_2=32\text{Hz}$ 的正弦波叠加而成，传输特性选择 32 阶非对称结构即 $\boldsymbol{H}=[1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1]$。

```
% this function is to testify the 1st equivalent all-phase filter
function apDFT1(N)
if nargin = =0
  N =32;
end
H = [1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1];
f1 =16;f2 =32;fs =max(f1,f2)* 10;
```

```
tstart = -N/fs;tend =2/min(f1,f2) + (N -1)/fs;
t =tstart:1/fs:tend;
s =  sin(2* pi* f1* t) +cos(2* pi* f2* t);
slow =min(s);shigh =max(s);siglen =length(t);
out0 =ap0(N,siglen,s,H);out1 =ap1(N,siglen,s,H);
subplot(2,2,1),plot(t(N:siglen -N +1),real(out0),'k','LineWidth',2);
hold on,plot(t(N:siglen -N +1),imag(out0),'k - -','LineWidth',2);
legend('real part','imaginary'),title('ap0');
grid on,axis([0 tend slow shigh]);
xlabel('t(s)'),ylabel('amplitude ');
subplot(2,2,2),plot(t(N :siglen -N + 1),real(out1),'k','LineWidth',2);
hold on,plot(t(N:siglen -N +1),imag(out1),'k - -','LineWidth',2);
legend('real part','imaginary'),title('ap1');
grid on,axis([0 tend slow shigh]);
xlabel('t(s)'),ylabel('amplitude ');
subplot(2,2,3),plot(t(N :siglen -N +1),abs(out0) -abs(out1),
'k','LineWidth',2);
grid on,axis([0 tend -0.1 0.1]);
xlabel('t(s)'),ylabel('amplitude frequency substract');
subplot(2,2,4),plot(t(N :siglen -N +1),angle(out0) -angle(out1),
'k','LineWidth',2);
grid on,axis([0 tend -0.1 0.1]);
xlabel('t(s)'),ylabel('angle frequency substract');
function out0 =ap0(N,siglen,s,H)
d =zeros(1,N);
for i =1 :siglen -N +1
    infft =fft(s(i:i +N -1));outfft =H.* infft;
    out =ifft(outfft);
    if(i >N -1)
        y(i) =(d(1) +out(1))/N;
    end
    for j =1:N -2
        d(j) =d(j +1) +out(j +1);
    end
    d(N -1) =out(N);
end
out0 =y(N :siglen -N +1);
function out1 =ap1(N,siglen,s,H)
h =ifft(H);hext =[h(2:N),h(1: N)];
triwin =triang(2* N -1);
hwined =hext.* triwin';
```

```
for i =N :siglen -N +1
    out1 (i -N +1) =sum(s(i +N -1: -1:i -N +1).* hwined);
end
```

运行结果如图 2-18 所示。

图 2-18　ap0 与 ap1 等效性验证

由图 2-17 可知，ap1 结构中 FIR 滤波器单位冲击响应 $\boldsymbol{h}=[h(1)h(2)\cdots h(N-1)h(0)h(1)\cdots h(N-2)h(N-1)]$。利用 $\boldsymbol{h}$ 的周期性将 ap1 又可简化为如图 2-19 所示。

图 2-19　N 阶无窗全相位滤波器第二等效图（ap2）

ap1 共用到了延迟单元 $2N-2$ 个，加法器 $2N-2$ 个，乘法器 $4N-2$ 个，但对比 ap0 省去了 DFT/IDFT 复数运算；ap2 使用延迟单元 $2N-2$ 个，加法器 $2N-2$ 个，乘法器 $3N-1$ 个，对比 ap1 节省乘法器 $N-1$ 个。设 $\boldsymbol{H}=[1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0]$，实现 ap2 的 MATLAB 代码如下：

```
function out2 =ap2 (N,siglen,s,H)
h =ifft(H);
triwin =triang(2* N -1)';
for i =N :siglen -N +1
    xwined =s(i +N -1: -1:i -N +1).* triwin;
    xwo =[xwined(N),xwined(1:N -1) +xwined(N +1:2* N -1)];
    out2 (i -N +1) =sum(xwo.* h);
end
```

运行结果如图 2-20 所示。

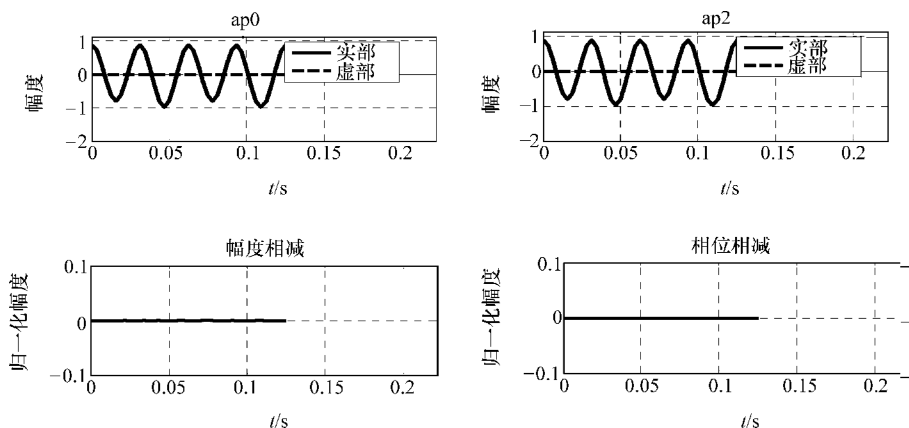


图 2-20 ap0 与 ap2 等效性验证

由 ap0 的结构可知，在任意时刻 n 的输入 $x(n)$ 都有对应的输出 $y(n)$，频率采样序列 $\boldsymbol{H}$ 具有类模拟系统传输特性。为充分发挥全相位数字滤波器在频域中的滤波作用，利用 DFT 可得到等效频域结构。令 $\boldsymbol{x}_{\mathrm{wo}}=[x_{\mathrm{wo}}(n)\,x_{\mathrm{wo}}(n-1)\cdots x_{\mathrm{wo}}(n-N+1)]^{\mathrm{T}}$，则

$$\boldsymbol{x}_{\mathrm{wo}}=\boldsymbol{O}\cdot\boldsymbol{C}\cdot\boldsymbol{x}_n \tag{2-32}$$

式中，$\boldsymbol{O}$ 和 $\boldsymbol{C}$ 分别称为重叠矩阵和卷积窗，它们有如下形式：

$$\boldsymbol{O}=\begin{pmatrix} 0 & 0 & \cdots & 0 & 1 & 0 & 0 & \cdots & 0\\ 1 & 0 & \cdots & 0 & 0 & 1 & 0 & \cdots & 0\\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\ 0 & \cdots & 1 & 0 & 0 & 0 & \cdots & 1 & 0\\ 0 & 0 & \cdots & 1 & 0 & 0 & 0 & \cdots & 1 \end{pmatrix}_{N\times(2N-1)},\quad \boldsymbol{C}=\mathrm{diag}(c_0\cdots c_{N-2}c_{N-1}c_{N-2}\cdots c_0) \tag{2-33}$$

无窗全相位数字滤波器的卷积窗元素为

$$c_k=\frac{k+1}{N}\quad k=0,1,\cdots,N-1 \tag{2-34}$$

图 2-19 中的输出可表示如下：

$$\begin{aligned} y(n)&=\boldsymbol{x}_{\mathrm{wo}}^{\mathrm{T}}\boldsymbol{h}=\frac{1}{N}\boldsymbol{x}_{\mathrm{wo}}^{\mathrm{T}}\boldsymbol{W}_N^{*}\ \boldsymbol{W}_N\\ \boldsymbol{h}&=\frac{1}{N^2}(\boldsymbol{W}_N\,\boldsymbol{x}_{\mathrm{wo}})^{\mathrm{T}}\cdot(\boldsymbol{W}_N^{*}\ \boldsymbol{W}_N^{*})\cdot\\ (\boldsymbol{W}_N\boldsymbol{h})&=\frac{1}{N}\boldsymbol{X}_{\mathrm{wo}}^{\mathrm{T}}\cdot\boldsymbol{H}_{\mathrm{r}} \end{aligned} \tag{2-35}$$

式（2-35）中，$\boldsymbol{X}_{\mathrm{wo}}$是向量$\boldsymbol{x}_{\mathrm{wo}}$的离散傅里叶变换，$\boldsymbol{H}_{\mathrm{r}}$ 由下式定义：

$$\boldsymbol{H}_{\mathrm{r}}=\frac{1}{N}\boldsymbol{W}_N^*\boldsymbol{W}_N^*\cdot\boldsymbol{H}=\begin{pmatrix}1&\cdots&0&0&0\\0&\cdots&0&0&1\\0&\cdots&0&1&0\\\vdots&\vdots&\vdots&\vdots&\vdots\\0&1&\cdots&0&0\end{pmatrix}\begin{pmatrix}H(0)\\H(1)\\H(2)\\\vdots\\H(N-1)\end{pmatrix}=\begin{pmatrix}H(0)\\H(N-1)\\H(N-2)\\\vdots\\H(1)\end{pmatrix}\tag{2-36}$$

式（2-35）就是全相位数字滤波器的频域表达式，其系统结构如图 2-21 所示。

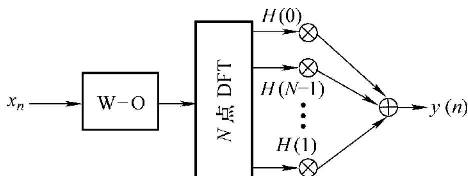


图 2-21　N 阶无窗全相位滤波器第三等效图（ap3）

与图 2-15 相比，只经过一次 DFT 但要对输入信号做加权重叠处理。此结构频域滤波器可以通过调节 $\boldsymbol{H}$ 来有效地控制信号各个频率分量，全相位数字滤波器具有传统频率采样法设计的特征，对原信号加权和重叠处理又兼有窗函数设计法特点。实现 ap3 的 MATLAB 代码(设 H = [1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0])如下：

```
function out3 =ap3(N,siglen,s,H)
triwin =triang(2* N -1)';
Hr =[H(1),H(N: -1:2)];
for i =N :siglen -N +1
    xwined =s(i +N -1: -1:i -N +1).* triwin;
    xwo =[xwined(N),xwined(1:N -1) +xwined(N +1:2* N -1)];
    xwofft =fft(xwo,N);
    out3(i -N +1) =sum(xwofft.* Hr)/N;
end
```

2.2.2　DFT 域单窗全相位数字滤波器

由上节内容可知，全相位数字滤波器兼有加窗和频域设计的双重优点，但没有改善通带和阻带内的振荡幅度大小，这可由图 2-16 的幅频特性曲线可以看出。实际上，在对全相位数字滤波器中的输入没有经过任何预先处理情况下的输出，是与 x_n相关的 N 个数据块分别延迟 $k(k=N-1,\ N-2,\ \cdots,\ 1,\ 0)$所产生的 N 个输出结果的算术平均，可以用下式表示：

$$
\begin{array}{cccccccc}
 & & & x(n) & x(n-1) & \cdots & x(n-N+2) & x(n-N+1) \\
 & & x(n+1) & x(n) & x(n-1) & \cdots & x(n-N+2) & \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
x(n+N-1) & \cdots & x(n+1) & x(n) & & & & \\
 & & & \Downarrow & & & & \\
1 & \cdots & N-1 & N & N-1 & \cdots & 2 & 1 \\
 & & & \times & & & & \\
x(n+N-1) & \cdots & x(n+1) & x(n) & x(n-1) & \cdots & x(n-N+2) & x(n-N+1) \\
 & & & \Downarrow & & & & \\
x(n+N-1) & \cdots & (N-1)x(n+1) & Nx(n) & (N-1)x(n-1) & \cdots & 2x(n-N+2) & x(n-N+1)
\end{array}
$$

这个全相位数字滤波器可等效为带有对称三角窗的 FIR 滤波器。若将输入先加权（即加窗后）再进行滤波，则等效的 FIR 中的卷积窗就会改变，从而改变滤波器特性。

全相位数字滤波器中加的窗序列可在 DFT 前或 IDFT 之后，即对时域信号加权。在 DFT 前加窗的系统如图 2-22 所示。

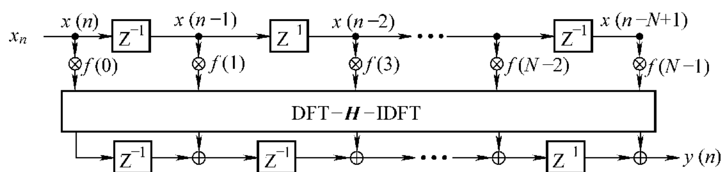


图 2-22　加前单窗的全相位数字滤波器

记前窗序列为

$$F=[f(N-1)\ \cdots\ f(1)\ f(0)] \tag{2-37}$$

与 2.2.1 节推导无窗全相位系统相同方法可得到输出为

$$y(n)=\frac{1}{N}\sum_{i=0}^{N-1}\left[\boldsymbol{s}_{N-1-i}^{\mathrm{T}}\cdot\boldsymbol{W}_N^{*}\cdot\boldsymbol{H}\cdot\boldsymbol{W}\cdot(\mathrm{diag}(\boldsymbol{F})\cdot\boldsymbol{x}_{n+i})\right] \tag{2-38}$$

将式（2-38）展开得：

$$
\begin{aligned}
y(n) &= \sum_{i=0}^{N-1}\left\{\sum_{j=0}^{N-1}[h(-i-1-j)x(n+i-N+1+j)f(N-1-j)]\right\}\\
&= \sum_{i=0}^{N-1}\left\{\sum_{k=-i}^{-i+N-1}[h(k)x(n-k)f(k+i)]\right\},\quad k=-i+N-1-j\\
&= \sum_{k=-N+1}^{N-1}\{[f(k)*R_{\mathrm{N}}(-k)][h(k)x(n-k)]\}\\
&= \sum_{k=-N+1}^{N-1}[C_{\mathrm{s}}(k)h(k)x(n-k)],C_{\mathrm{s}}(k)=f(k)*R_{\mathrm{N}}(-k)\\
&= \sum_{k=-N+1}^{N-1}[h_{\mathrm{c}}(k)x(n-k)]\\
&= h_{\mathrm{c}}(n)*x(n)
\end{aligned} \tag{2-39}
$$

式中，$\boldsymbol{C}_s$ 是卷积窗序列，由图 2-22 中的前窗序列与长度为 N 的矩形窗 $\boldsymbol{R}_N$ 自身镜像反转后卷积形成，冲击响应序列 $\boldsymbol{h}_c(n)=\boldsymbol{h}(n)\cdot\boldsymbol{C}_s(n)$。由上面结果可以看出，对于加前窗 $\boldsymbol{F}$ 的全相位数字滤波器也等效于 $2N-1$ 阶带窗 FIR 滤波器，窗序列是由 $\boldsymbol{F}$ 和矩形窗卷积产生，系统的单位冲击响应等于 $\boldsymbol{H}$ 的反离散傅里叶变换的两个周期延拓。因此，对比无窗全相位数字滤波器的唯一区别就是卷积窗的大小和形状。

需要指出的是，为保证输出 $y(n)$ 幅度的正确性，需对卷积窗进行归一化处理。观察式（2-39），只需卷积窗所有元素之和等于 N 就可以保证幅度不失真，即

$$\sum_{k=-N+1}^{N-1}[\boldsymbol{C}_s(k)]=N \tag{2-40}$$

单窗全相位滤波器的归一化充分条件是前窗矢量的 1 – 范数等于 1。对于具有线性相位的单窗系统，式（2-40）等价于

$$\boldsymbol{C}_s(0)=\sum_{n=0}^{N-1}f(n)=\|\boldsymbol{f}\|_1=1 \tag{2-41}$$

在后面介绍的全相位性质可知式（2-41）也是充分条件。实现单窗系统的 MATLAB 代码如下：

```
% this function performs to draw out the amplitude frequency
% with consin input for all phase digital filter with single window
function out =apSWF(N)
clc,clf;
if nargin = =0
  N =32;
  samlen =N* 8;
end
% transmission property
H =[1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1];
% frequency sampling array
w = -pi:2* pi /samlen:pi;
t = -N +1 :samlen;s =cos(w'* t);
Hs =repmat(H,size(w',1),1);d =zeros(size(w',1),N);
% triangle window
fwin1 =triang(N)';fwin1 =fwin1/sum(fwin1);
fwins(:,:,1) =repmat(fwin1,size(w',1),1);
fwin2 =boxcar(N)';fwin2 =fwin2/sum(fwin2);
fwins(:,:,2) =repmat(fwin2,size(w',1),1);
% filtering process
for k =1:2
```

```
    for i =1 :samlen -N +1
        sin =s(:,i:i +N -1).* fwins(:,:,k);
        infft =fft(sin,[],2);outfft =Hs.* infft;
        out =ifft(outfft,[],2);
        if(i >N -1)
            y(:,i) =abs(d(:,1) +out(:,1));
        end
        for j =1:N -2
            d(:,j) =d(:,j +1) +out(:,j +1);
        end
        d(:,N -1) =out(:,N);
    end
    % draw amp/freq and phase/freq curve
    yy(:,k) =max(y,[],2);
end

subplot(1,2,1),h1 =plot(w,yy,'k','LineWidth',2);grid on,axis tight;
set(h1(2),'LineStyle','- -');legend('triangle','rectangle');
xlabel('\omega(rad/s)'),ylabel('amplitude ');title('amp - freq curve');
subplot(1,2,2),h2 =plot(w,20* log10(yy),'k','LineWidth',2);
set(h2(2),'LineStyle','- -');legend('triangle','rectangle');
xlabel('\omega(rad/s)'),ylabel('attenuation');
title('phase - freq curve');grid on,axis([0 pi -100 0]);
```

此代码画出了加三角窗时的传输特性曲线，并与无窗时的特性进行了对比，结果如图 2-23 所示。从图中可看出，单窗全相位数字滤波器极大地改善了通带及阻带内的波纹数量且降低了波纹幅度，最小旁瓣衰减也降低了 6. 8dB。

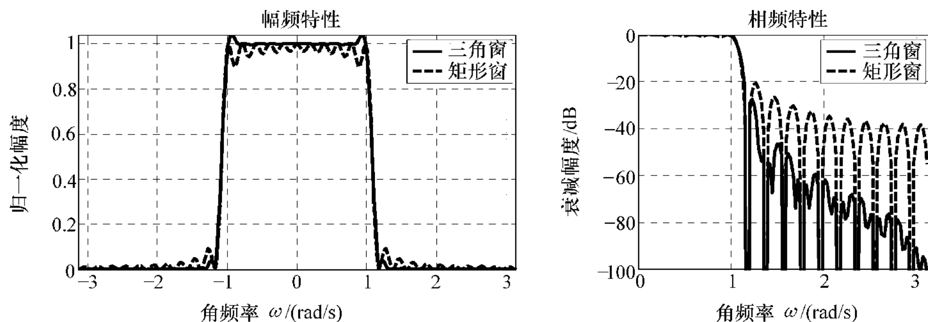


图 2-23 加前单窗对比无窗全相位数字滤波器特性

在 MATLAB 代码中可以设置不同的窗函数来调整滤波器特性，设置不同的 ***H*** 来对信号进行滤波。输入信号由频率 16Hz 和 32. 1Hz 两正弦波构成，滤波器

阶数为 32，频率分辨率为 16Hz/s，设定 $\boldsymbol{H}=[0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0]$ 以恢复得到 32.1Hz 正弦波。分别用无窗和单窗全相位滤波方法得到的结果和误差如图 2-24 所示（误差平均值分别为 8.0441e－006、6.1618e－006）。

图 2-24　无窗和单窗全相位数字滤波结果对比

图 2-22 中的窗在 DFT 变换之前加入，谱为信号和窗函数谱的卷积而并非信号的实际谱。如果将窗加在 IDFT 之后，得到加后单窗全相位系统如图 2-25 所示。

图 2-25　加后单窗的全相位数字滤波器

记后窗序列为

$$\boldsymbol{B}=[\,b(N-1)\quad \cdots \quad b(1)\quad b(0)\,] \tag{2-42}$$

与 2.2.1 节推导无窗全相位系统相同方法可得到输出为

$$y(n)=\frac{1}{N}\sum_{i=0}^{N-1}\left[\boldsymbol{s}_{N-1-i}^{\mathrm{T}}\cdot \mathrm{diag}(\boldsymbol{B})\cdot \boldsymbol{W}_N^{*}\cdot \boldsymbol{H}\cdot \boldsymbol{W}\cdot \boldsymbol{x}_{n+i}\right] \tag{2-43}$$

将上式（2-43）展开得：

$$
\begin{aligned}
y(n) &= \sum_{i=0}^{N-1}\left\{b(i)\sum_{j=0}^{N-1}[h(-i-1-j)x(n+i-N+1+j)]\right\}\\
&= \sum_{i=0}^{N-1}\left\{b(i)\sum_{k=-i}^{-i+N-1}[h(k)x(n-k)]\right\},\quad k=-i+N-1-j\\
&= \sum_{k=-N+1}^{N-1}\{[b(N-1-k)*R_{\mathrm{N}}(k+N-1)][h(k)x(n-k)]\} \qquad (2\text{-}44)\\
&= \sum_{k=-N+1}^{N-1}[C_{\mathrm{s}}(k)h(k)x(n-k)],\quad C_{\mathrm{s}}(k)=b(k)*R_{\mathrm{N}}(-k)\\
&= \sum_{k=-N+1}^{N-1}[h'(k)x(n-k)]\\
&= h'(n)*x(n)
\end{aligned}
$$

无论是带前窗还是后窗的全相位数字滤波器都等效于 FIR 滤波器，该滤波器的卷积窗为全相位结构中前窗或后窗与镜像的矩形窗卷积产生，通过调解窗可以改善系统的传输特性，单位冲击响应为频率采样序列 $\boldsymbol{H}$ 反 IDFT 变换的两个周期延拓。

2.2.3 DFT 域双窗全相位数字滤波器

为进一步改善系统性能，充分发挥窗函数在滤波器设计中的作用，综合上面两种情形得到在 DFT 前和 IDFT 后均加窗的全相位数字滤波器结构如图 2-26 所示。

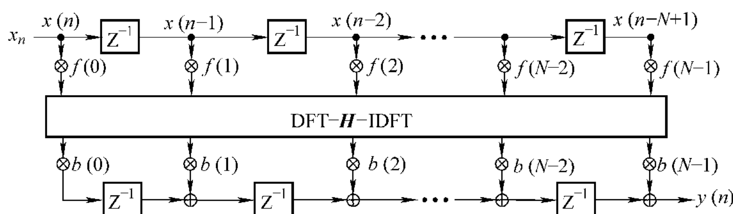


图 2-26 加双窗的全相位数字滤波器结构

在前窗 $\boldsymbol{F}$ 和后窗 $\boldsymbol{B}$ 的作用下，图 2-26 在 n 时刻的输出为

$$
\begin{aligned}
y(n) &= \frac{1}{N}\sum_{i=0}^{N-1}\left[\mathrm{s}_{N-1-i}^{\mathrm{T}}\cdot\mathrm{diag}(\boldsymbol{B})\cdot\boldsymbol{W}^{*}\cdot\boldsymbol{H}\cdot\boldsymbol{W}\cdot(\mathrm{diag}(\boldsymbol{F})\cdot\boldsymbol{x}_{n+i})\right]\\
&= \sum_{i=0}^{N-1}\left\{b(i)\sum_{k=-i}^{-i+N-1}[h(k)x(n-k)f(k+i)]\right\},\quad k=-i+N-1-j\\
&= \frac{1}{N}\left[b(0)\sum_{k=0}^{N-1}f(k)+\cdots+b(N-1)\sum_{k=-N+1}^{0}f(k+N-1)\right][h(k)x(n-k)]
\end{aligned}
$$

$$
\begin{aligned}
&= \frac{1}{N}\sum_{k=-N+1}^{N-1}\{[b(N-1-k)*f(k+N-1)][h(k)x(n-k)]\} \\
&= \sum_{k=-N+1}^{N-1}[C_d(k)h(k)x(n-k)],\quad C_d(k)=b(-k)*f(k) \\
&= \sum_{k=-N+1}^{N-1}h'(k)x(n-k) \\
&= h'(n)*x(n)
\end{aligned} \tag{2-45}
$$

卷积窗 $\boldsymbol{C}_d$ 由前窗与后窗镜像后卷积产生，通过调节 $\boldsymbol{F}$ 和 $\boldsymbol{B}$ 来改善系统性能，灵活性和实用性都好于单窗。双窗全相位数字滤波器的归一化充分条件也满足式（2-40），对于线性相位系统的等价条件如下：

$$
\sum_{k=0}^{N-1}\left[\sum_{n=0}^{k}f(n)g(N-k+n)+\sum_{n=k}^{N-1}f(n)g(n-k)-f(k)g(N)\right]=N \tag{2-46}
$$

对比无窗、单窗和双窗全相位数字滤波器特性的 MATLAB 实现（$N=32$）如下：

```
% this function performs to draw out the amplitude frequency with consin
% input for all phase digital filter under none/one/two window
function out =apFNSD(N)
clc,clf;
if nargin = =0
  N =32;samlen =N* 8;
end
H =[1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1];
w = -pi:2* pi /samlen:pi;t = -N +1:samlen;s =cos(w'* t);
Hs =repmat(H,size(w',1),1);d =zeros(size(w',1),N);
fwin1 =boxcar(N)';fwin1 =fwin1/sum(fwin1);
fwins(:,:,1) =repmat(fwin1,size(w',1),1);
fwin2 =triang(N)';fwin2 =fwin2/sum(fwin2);
fwins(:,:,2) =repmat(fwin2,size(w',1),1);
fwinf =triang(N)';fwinb =hanning(N)';
gyh =sum(fwinf.* fwinb);fwinf =fwinf/gyh;
fwins(:,:,3) =  repmat(fwinf,size(w',1),1);
fwins(:,:,4) =  repmat(fwinb,size(w',1),1);
% filtering process
for k =1:3
    for i =1 :samlen -N +1
        sin =s(:,i:i +N -1).* fwins(:,:,k);
        infft =fft(sin,[],2);outfft =Hs.* infft;
        out =ifft(outfft,[],2);
        if k = =3
```

```
            out = out.* fwins(:,:,k +1);
        end
        if(i > N -1)
            y(:,i) = abs(d(:,1) + out(:,1));
        end
        for j =1:N -2
            d(:,j) = d(:,j +1) + out(:,j +1);
        end
        d(:,N -1) = out(:,N);
    end
    yy(:,k) = max(y,[],2);
end
subplot(1,2,1),h1 = plot(w,yy,'k','LineWidth',2.8);grid on,axis tight;
set(h1(2),'LineStyle','- -');set(h1(3),'LineStyle',':');
legend('none window','single window','two windows');
xlabel('\omega(rad/s)'),ylabel('amplitude ');title('amp - freq curve');
subplot(1,2,2),h2 = plot(w,20* log10(yy),'k','LineWidth',2.8);
set(h2(2),'LineStyle','- -');set(h2(3),'LineStyle',':');
legend('none window','single window','two windows');
xlabel('\omega(rad/s)'),ylabel('attenuation');
title('phase - freqcurve');grid on,axis([0 pi  -100 0]);
out = [yy];
```

运行结果如图 2-27 所示。可以看出，双窗系统在通带和阻带内的波纹基本消失，且对比单窗在过渡带的吉布斯脉冲也消失，第一旁瓣衰减比单窗降低 50dB，具有十分优良的特性。

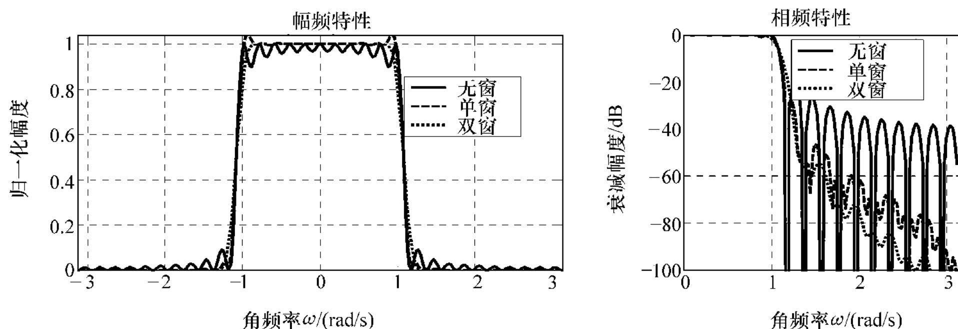


图 2-27　无窗、单窗和双窗全相位数字滤波器特性对比

分别用单窗和双窗全相位滤波方法对由频率为 16Hz 和 48.4Hz 两正弦波构成的输入信号进行滤波实验，滤波器阶数为 32，频率分辨率为 16Hz/s，设定 $\boldsymbol{H}$ = [0 0 0100 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0] 以恢复得到 48.4Hz

正弦波。其 MATLAB 代码如下：

```
% this function is to compare the filtering result by means of one and two
% windows all-phase filter
function apDS(N)
clc,clf;
if nargin = =0
  N =32;
end
H =[0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0];
f1 =16;f2 =48.4;fs =N* 16;
tstart = - (N -1)/fs;tend =1/floor(min(f1,f2)) + (N -1)/fs;
t =tstart:1/fs:tend;s1 =sin(2* pi* f1* t);
s2 =cos(2* pi* f2* t);s =  s1 +s2;siglen =length(t);
out0 =apS(N,siglen,s,H);out1 =apD(N,siglen,s,H);
tlen =N:siglen -N +1;
subplot(2,2,1),plot(t(tlen),real(out0),'k','LineWidth',3);
hold on,plot(t(tlen),imag(out0),'k - -','LineWidth',3);
legend('real','imaginary'),title('ONE WINDOW');
grid on,axis tight;xlabel('t(s)'),ylabel('amplitude ');
subplot(2,2,2),plot(t(tlen),real(out1),'k','LineWidth',3);
hold on,plot(t(tlen),imag(out1),'k - -','LineWidth',3);
legend('real','imaginary'),title('TWO WINDOWS');
grid on,axis tight;xlabel('t(s)'),ylabel('amplitude ');
subplot(2,2,3),plot(t(N:siglen -N +1),out0 -s2(tlen),'k','LineWidth',3);
grid on,axis([0 t(siglen -N +1) -0.01 0.01]);
xlabel('t(s)'),ylabel('errors');mean(abs(out0 -s2(tlen)))
subplot(2,2,4),plot(t(N:siglen -N +1),out1 -s2(tlen),'k','LineWidth',3);
grid on,axis([0 t(siglen -N +1) -0.01 0.01]);
xlabel('t(s)'),ylabel('errors');mean(abs(out1 -s2(tlen)))
function out1 =apD(N,siglen,s,H)
d =zeros(1,N);fwin =triang(N)';bwin =chebwin(N)';
gyh =sum(conv(fwin,bwin))/N;fwin =fwin/gyh;
for i =1 :siglen -N +1
    infft =fft(s(i:i +N -1).* fwin);
    outfft =H.* infft;out =ifft(outfft).* bwin;
if(i >N -1)
      y(i) =d(1) +out(1);
end
for j =1:N -2
  d(j) =d(j +1) +out(j +1);
end
```

```
  d(N-1)=out(N);
end
out1=y(N:siglen-N+1);
function out0=apS(N,siglen,s,H)
d=zeros(1,N);fwin=triang(N)';fwin=fwin/sum(fwin);
for i=1:siglen-N+1
    infft=fft(s(i:i+N-1).*fwin);
    outfft=H.*infft;out=ifft(outfft);
if(i>N-1)
      y(i)=d(1)+out(1);
end
for j=1:N-2
      d(j)=d(j+1)+out(j+1);
end
    d(N-1)=out(N);
end
out0=y(N:siglen-N+1);
```

得到的结果和误差如图 2-28 所示（误差平均值分别为 0.0010、4.9608e-004）。

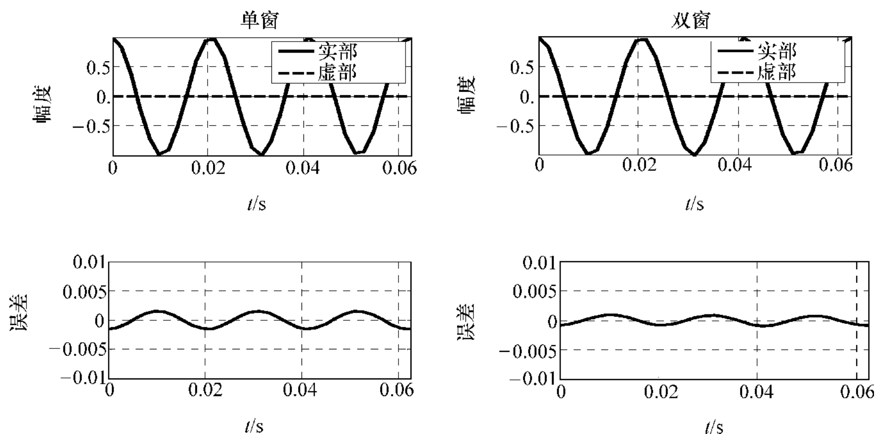


图 2-28　加双窗对比单窗滤波结果对比

上面介绍的 3 种带窗全相位数字滤波器具有完全相同的 FIR 滤波器等价结构，区别仅仅在于卷积窗的生成方法。实际上，对于“不加窗”的情况可视为前后都加矩形窗，因而 4 种类型均为带双窗系统。只是在第 1 种类型中前后都是矩形窗，第 2 种类型后窗是矩形窗，第 3 种类型前窗是矩形窗，第 4 种类型前后窗都不是矩形窗。为了便于讨论，将第 1 种类型称无窗全相位数字滤波器（all phase filter without Window，apFW），第 2 种和第 3 种类型称单窗全相位数字滤波

器（all phase filter with Single Window，apSW），第 4 种类型称双窗全相位数字滤波器（all phase filter with Dual Windows，apDW）。apNW、apSW 和 apDW 对应的卷积窗分别记为 $\boldsymbol{C}_{\mathrm{n}}$、$\boldsymbol{C}_{\mathrm{s}}$ 和 $\boldsymbol{C}_{\mathrm{d}}$，生成方式和归一化条件如下：

$$
\begin{aligned}
C_{\mathrm{n}}(n) &= R_{\mathrm{N}}(n) * R_{\mathrm{N}}(-n) \quad R_{\mathrm{N}}(n) = 1/N \\
C_{\mathrm{s}}(n) &= f_{\mathrm{N}}(n) * R_{\mathrm{N}}(-n) \quad \sum_{n=0}^{N-1} f_{\mathrm{N}}(n) = 1 \\
C_{\mathrm{d}}(n) &= f_{\mathrm{N}}(n) * b_{\mathrm{N}}(-n) \quad \sum_{n=-N+1}^{N-1} C_{\mathrm{d}}(n) = N
\end{aligned}
\tag{2-47}
$$

与 apNW、apSW 和 apDW 等价的 FIR 滤波器结构则可以表示如图 2-29 所示。

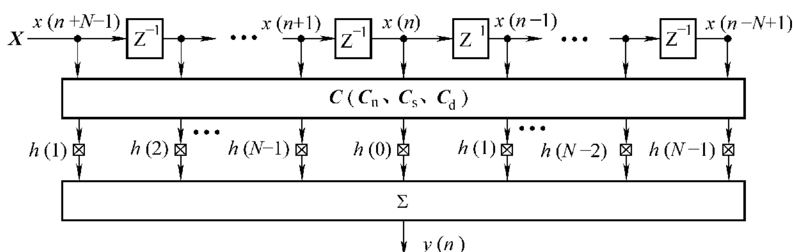


图 2-29　apNW、apSW 和 apDW 对应的 FIR 滤波器

2.3　全相位数字滤波器特性

全相位数字滤波器针对输入的信号进行了 $N-1/N$ 最大程度重叠，每单位抽样时刻的输入都遍历单位圆上所有相位得到对应的输出，而使频率抽样序列 $\boldsymbol{H}$ 更准确地控制系统传输特性。如图 2-30 所示“全相”的示意。

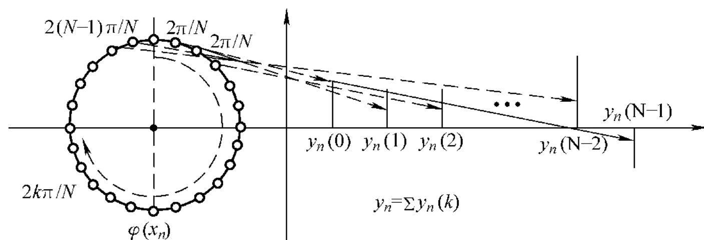


图 2-30　“全相”的含义

由等效图 2-29 可知，单位冲击响应为准 2 倍周期延拓。对于线性相位系统，则同时满足三种中心对称，即 $h(0)$ 中心对称、$h(0)$ 左右两侧满足 $h(N-k)=h(k)$ 的对称。加窗使全相位数字滤波器得到极大的改善，卷积窗的设计影响滤波器特性，对于线性相位要求窗必须满足一定的条件。由图 2-21 看出，对输入信

号进行加权重叠(W-O)是新颖的处理方法，它改变了信号某方面特性（如自相关）等，为全相位滤波器特性改善奠定了数据基础。以上这些特征使全相位数字滤波器具有很多重要性质。

定理 1：N 阶 ap 系统具有线性相位的充要条件是：$H(k)=H(N-k)$，且对于单窗系统要求非矩形窗中心对称，对于双窗系统要求 $\boldsymbol{F}=\boldsymbol{B}$ 或者前后窗均中心对称。

充分性证明：由式（2-45）容易推导出 ap 系统传输特性如下：

$$\begin{aligned}H(e^{j\omega}) &= \sum_{n=-N+1}^{N-1} C(n)h(n)e^{-jn\omega} \\ &= C(0)h(0) + \sum_{n=1}^{N-1} C_n h(n)e^{-jn\omega} + \sum_{n=1}^{N-1} C_{-n}h(N-n)e^{jn\omega}\end{aligned} \tag{2-48}$$

因为 $H(k)=H(N-k)$，所以 $h(k)=h(N-k)$；对于单窗系统 $f(k)=f(N-1-k)$，有

$$C(-k)=f(-k)*R_N(k)=f(N-1-k)*R_N(k+N-1)=f(k)*R_N(-k)=C(k) \tag{2-49}$$

对于双窗系统若 $\boldsymbol{F}=\boldsymbol{B}$，显然满足 $C(k)=C(-k)$；若前后窗均满足中心对称，则有

$$C(-k)=f(-k)*b(k)=f(N-1-k)*b(k+N-1)=f(k)*b(-k)=C(k) \tag{2-50}$$

故式（2-48）可以简化为

$$H(e^{j\omega}) = C(0)h(0) + 2\sum_{n=1}^{N-1} C(n)h(n)\cos n\omega = H_g(\omega) \tag{2-51}$$

即系统具有严格零相位。若 $\omega_k(0\leqslant k\leqslant N-1)$ 表示介于 $[0,2\pi]$ 的频率，并记矩阵：

$$\boldsymbol{COS}=\begin{bmatrix}1 & \cos(\omega_0) & \cos(2\omega_0) & \cdots & \cos[(N-1)\omega_0] \\ 1 & \cos(\omega_1) & \cos(2\omega_1) & \cdots & \cos[(N-1)\omega_1] \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \cos(\omega_{N-2}) & \cos(2\omega_{N-2}) & \cdots & \cos[(N-1)\omega_{N-2}] \\ 1 & \cos(\omega_{N-1}) & \cos(2\omega_{N-1}) & \cdots & \cos[(N-1)\omega_{N-1}]\end{bmatrix}_{N\times N},$$

$$\boldsymbol{CH}=\begin{bmatrix}C(0)h(0) \\ 2C(1)h(1) \\ \vdots \\ 2C(N-2)h(N-2) \\ 2C(N-1)h(N-1)\end{bmatrix}_{N\times 1},\ \boldsymbol{HG}=\begin{bmatrix}H(\omega_0) \\ H(\omega_1) \\ \vdots \\ H(\omega_{N-2}) \\ H(\omega_{N-1})\end{bmatrix}_{N\times 1} \tag{2-52}$$

则式（2-51）可以用矩阵表示如下：

$$\boldsymbol{COS} \cdot \boldsymbol{CH} = \boldsymbol{HG} \tag{2-53}$$

必要性证明：将具有线性相位的 ap 系统传输特性写成 $H(\mathrm{e}^{\mathrm{j}\omega}) = H_g(\omega)\mathrm{e}^{\mathrm{j}\varphi(\omega)}$，并令相位特性 $\varphi(\omega) = -\omega\tau$。记 $c_n = C(n)$，将（2-47）式展开并利用 $\boldsymbol{h}$ 的周期性整理得

$$\varphi(\omega) = -\arctan\frac{\sum_{n=-N+1}^{N-1} c_n h(n)\sin(n\omega)}{\sum_{n=-N+1}^{N-1} c_n h(n)\cos(n\omega)} = -\omega\tau \Rightarrow \sum_{n=-N+1}^{N-1}[c_n h(n)\sin n(\omega-\tau)] = 0 \tag{2-54}$$

$\tau = 0$ 时的系统具有严格零线性相位，正弦函数关于 $n = 0$ 处奇对称，所以上式成立的条件是 $c_n h(n)$ 关于 $n = 0$ 偶对称，即下式对于各种系统恒成立：

$$c_n h(n) = c_{-n} h(-n) \Rightarrow \begin{cases} c_n = c_{-n} \\ h(n) = h(-n) \end{cases} \quad n \in [0, N-1] \tag{2-55}$$

定理 1 表明，为保证加窗 ap 系统具有严格零相位，除要求频率抽样序列 $\boldsymbol{H}$ 具有对称性以外，还要对双窗做出限制：apSW 前/后窗具有中心对称性，apDW 前后窗相等。在后面的讨论中一般针对线性系统，除非特别指出。按照式（2-53）方式生成 ap 的 MATLAB 代码如下：

```
function apF(N,wp)
clc,clf;
if nargin = =0
  N =32;wp =pi/2;
end
Hp =zeros(1,N/2 +1);nones =floor((wp/pi)* N/2);
Hp(1:nones) =1;H =[Hp,fliplr(Hp(2:end -1))];
h =ifft(H);w =0:pi/(N* N):pi;cosw =w'* [0:N -1];
wb =hanning(N);wf =triang(N);wconv =conv(wb,wf);
C =wconv/wconv(N);cn =C(N:2* N -1);
hg =sum(2* cos(cosw)* diag(cn.* h'),2);
hg =hg -C(N)* h(1);
subplot(1,2,1),plot(w,hg,'k','LineWidth',3),grid on;
xlabel('\omega(rad/s)');ylabel('normalized amplitude');
subplot(1,2,2),plot(w,20* log10(abs(hg)),'k','LineWidth',3),
grid on;xlabel('\omega(rad/s)');ylabel('attenuation(dB)');
```

运行结果如图 2-31 所示。

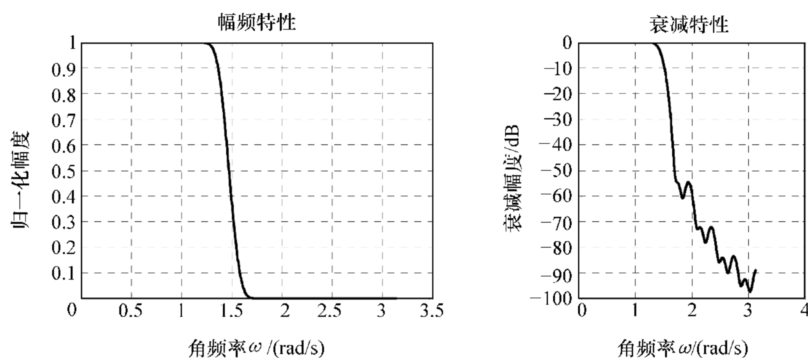


图 2-31 按式（2-52）构造的线性 apDW

定理 2：N 阶线性 apNW 和 apSW 系统满足 $H(e^{j\omega})|_{\omega=2n\pi/N}=H(n)$ 的充要条件是卷积窗1－范数恒等于 N，即满足 $\|\boldsymbol{C}\|_1=N$；apDW 则满足 $H(e^{j\omega})|_{\omega=2n\pi/N}<H(n)$。

证明：ap 系统的传输特性可表示如下：

$$
\begin{aligned}
H(e^{j\omega}) &= \sum_{n=-N+1}^{N-1} c_n h(n) e^{-jn\omega} \\
&= c_0 h(0) + \sum_{n=1}^{N-1}\left[c_n h(n) e^{-jn\omega} + c_{-N+n} h(-N+n) e^{-jn\omega}\right] \\
&= c_0 h(0) + \sum_{n=1}^{N-1}\left[c_n + c_{N-n}\right] h(n) e^{-jn\omega}
\end{aligned}
\tag{2-56}
$$

对于 apNW 和 apSW 的卷积窗，有

$$
\begin{aligned}
c_n &= f(n) * R_N(-n) = \sum_{k=n}^{N-1} f(k) \quad n \in [0, N-1] \\
c_{N-n} &= \sum_{k'=N-n}^{N-1} f(k') \overset{k=N-1-k'}{=} \sum_{k=0}^{n-1} f(k) \quad n \in [1, N-1]
\end{aligned}
\tag{2-57}
$$

把式（2-57）代入到式（2-56）得到

$$
\begin{aligned}
H(e^{j\omega}) &= c_0 h(0) + \sum_{k=0}^{N-1} f(k) \sum_{n=1}^{N-1} h(n) e^{-jn\omega} \\
&= \sum_{k=0}^{N-1} f(k) \sum_{n=0}^{N-1} h(n) e^{-jn\omega} \\
&= H(e^{j\omega}) \sum_{k=0}^{N-1} f(k)
\end{aligned}
\tag{2-58}
$$

$$
\Leftrightarrow \sum_{k=0}^{N-1} f(k) = 1 \Leftrightarrow \|\boldsymbol{C}\|_1 = N
$$

对于 apNW 和 apSW 窗序列 $\|\boldsymbol{f}\|_1=1$ 就能满足定理 2 的条件。但是，对于 apDW 的卷积窗元素：

$$
\begin{aligned}
c_n &= \sum_{k=n}^{N-1} f(k)b(k-n) \quad n=0,1,\cdots,N-1 \\
c_{N-n} &= \sum_{k'=N-n}^{N-1} f(k')b(k'-N+n) \\
&\overset{k=N-1-k'}{=} \sum_{k=0}^{n-1} f(N-1-k)b(n-1-k), \\
n &= 1,\cdots,N-1
\end{aligned}
\tag{2-59}
$$

定义频偏向量如下：

$$
c_n + c_{N-n} = \boldsymbol{E}_n \tag{2-60}
$$

式中，$\boldsymbol{E}_n$ 反映了系统特性在频率取样点的偏离程度，只要 $\boldsymbol{E}_n$ 具有恒定值就可保证两者相等，而 $\boldsymbol{E}_n$ 波动越大代表特性曲线偏离采样点程度越大。apDW 的卷积窗元素之和显然为 n 的函数，为使 $\boldsymbol{E}_n$ 平稳而应满足条件：

$$
\begin{aligned}
f(k) &= f(N-1-k) \\
\boldsymbol{f} &= \boldsymbol{b}
\end{aligned}
\tag{2-61}
$$

即要求前后窗相当且具有中心对称特性。结合式（2-61）和式（2-59）可得：

$$
\boldsymbol{E}_n = \begin{cases} \displaystyle\sum_{k=0}^{N-1} f^2(k) & n=0 \\ \displaystyle\sum_{k=1}^{N-1} f(k)f(k-n) & n \in [1, N-1] \end{cases} \tag{2-62}
$$

$\boldsymbol{E}_n$ 关于中心对称，最大值在 $n=0$ 时获得且等于前窗的自相关。由于窗序列 $\boldsymbol{f}$ 中心能量最大，在对称区间内 $\boldsymbol{E}_n$ 单调递减，故用 $\boldsymbol{E}_0$ 归一化后的 $\boldsymbol{E}_n$ 只在 $n=0$ 处等于 1，其余各点都低于 1，即 $H(e^{j\omega})|_{\omega=2n\pi/N} < H(n)$。

定理 3：N 阶 ap 系统频率响应函数等于频率采样向量与卷积窗的离散卷积。即频率响应函数可通过频率采样序列内插实现，其内插函数为卷积窗谱函数。

证明：由式（2-48）可得到：

$$
\begin{aligned}
H(e^{j\omega}) &= \sum_{n=-N+1}^{N-1} C(n)\left[\frac{1}{N}\sum_{k=0}^{N-1} H(k)e^{j2\pi kn/N}\right]e^{-jn\omega} \\
&= \frac{1}{N}\sum_{k=0}^{N-1} H(k) \sum_{n=-N+1}^{N-1} C(n)e^{-jn(\omega-2\pi k/N)} \\
&\overset{\text{令}\Phi(j\omega)=\sum_{n=-N+1}^{N-1} C(n)e^{-jn\omega}}{=} \frac{1}{N}\sum_{k=0}^{N-1} H(k)\Phi[j(\omega-2k\pi/N)]
\end{aligned}
\tag{2-63}
$$

对于 N 阶传统频率采样法得到的频率响应函数为

$$H(e^{j\omega}) = \sum_{k=0}^{N-1} H(k)\Phi_T\left[j\left(\omega - \frac{2\pi}{N}k\right)\right]$$
$$\Phi_T(j\omega) = e^{-j\omega(N-1)/2} \cdot \frac{\sin(\omega N/2)}{N\sin(\omega/2)} \tag{2-64}$$

apNW 的前后窗均为长度为 N 的矩形窗，其卷积窗频谱如式（2-64）即 apNW 卷积窗谱 Φ_N（jω）是 Φ_T（jω）的幅频二次方，使旁瓣能量进一步降低而主瓣的能量占比提高。

$$\Phi_N(j\omega) = \Phi_T^2(j\omega) = \left\{\frac{\sin(\omega N/2)}{N\sin\omega/2}\right\}^2 \tag{2-65}$$

MATLAB 实现两种谱插值函数代码如下：

```
% this function is to draw the interpolation spectrum function
function c2_32(N)
clc,clf;
if nargin = =0
  N =32;udlim =4;np =0;
end
w = -udlim* pi/N:pi/N^2:udlim* pi/N;
for k = -np:np
    walias =2* pi/N* k;
    fi1(k +np +1,:) =sin(N* (w -walias))./...
        sin((w -walias))/N;
    fi2(k +np +1,:) = (sin(N* (w -walias))./...
    sin((w -walias))/N).^2;
end
h =plot(w,fi1,'k -o',w,fi2,'k -..','MarkerSize',6);grid on;
hlg =legend(h,'Traditional','apNW');
set(hlg,'FontSize',16);xlabel('omega(rad/s)','FontSize',16);
ylabel('amplitude','FontSize',16);
title('frequency interpolation function','FontSize',16);
```

运行结果如图 2-32 所示。从图 2-32 可以看出，apNW 的卷积窗谱具有以下

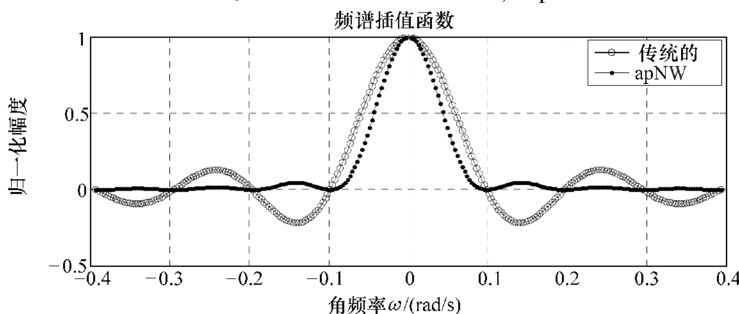


图 2-32　apNW 与频率采样频谱插值函数对比图

几个特点：一是谱为实数，由式（2-47）可知是矩形窗谱共轭相乘的结果，因此由式（2-64）内插得到的 apNW 系统谱也为实数；二是谱为非负数，且旁瓣幅度衰减速率随偏离距离增大而加剧，主瓣占比相对于 $\Phi_{\mathrm{T}}(\mathrm{j}\omega)$ 提升 68%；三是在频率采样点 $2k'\pi/N$ 处：

$$\Phi_{\mathrm{N}}[\mathrm{j}(\omega-2k\pi/N)]\big|_{\omega=\mathrm{j}\cdot 2k'\pi/N}=\begin{cases}1 & k'=k\\ 0 & k'\neq k\end{cases}\tag{2-66}$$

即 apNW 幅频曲线通过 $\boldsymbol{H}$ 的频率抽样点，采样点之间的幅度由式（2-64）确定插值函数确定。对于 N（奇数为例）阶 apSW，其归一化前窗 $\boldsymbol{f}$ 可以表示为

$$\boldsymbol{f}=\sum_{n=0}^{(N-1)/2}\frac{[f(n)-f(n+1)]}{\sum_{n=0}^{N-1}f(n)}R_{2n+1}(n)\tag{2-67}$$

假设边界点 $f[(N-1)/2+1]=0$，窗 $\boldsymbol{f}$ 可以表示 $(N-1)/2+1$ 个变长矩形窗之和，如图 2-33 所示。

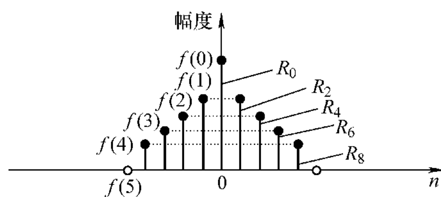


图 2-33　apSW 前窗矩形基分解图

由图 2-33 可以看出，幅度和长度均可变的矩形窗序列 $\boldsymbol{R}_{2n+1}$ 中各元素相互正交，即 $\{R_{2n+1}(n)\}$ 构成非完备正交系。apSW 卷积窗谱可表示如下：

$$\begin{aligned}\Phi_{\mathrm{S}}(\mathrm{j}\omega)&=\left[\frac{\sin(N\omega/2)}{N\sin(\omega/2)}\right]\cdot\sum_{n=0}^{(N-1)/2}\frac{[f(n)-f(n+1)]}{\sum_{n=0}^{N-1}f(n)}\left[\frac{\sin[(2n+1)\omega/2]}{\sin(\omega/2)}\right]\\&=\sum_{n=0}^{(N-1)/2}\frac{[f(n)-f(n+1)]}{\sum_{n=0}^{N-1}f(n)}\left[\frac{\sin[(2n+1)\omega/2]}{\sin(\omega/2)}\right]\left[\frac{\sin(N\omega/2)}{N\sin(\omega/2)}\right]\end{aligned}\tag{2-68}$$

apSW 卷积窗谱是 N 阶矩形窗谱与若干可变长度和幅度的矩形窗谱的积之和。显然，由于增加零值点且可变矩形窗谱的幅度更低而使旁瓣能量比 apNW 更低，主瓣的能量占比进一步提高；第一零值点仍为 $2\pi/N$，所以过渡带与频率采样法和 apNW 相同。在频率采样点 $2k'\pi/N$ 处，有

$$\begin{aligned}\Phi_{\mathrm{S}}[\mathrm{j}(\omega-2k\pi/N)]\big|_{\omega=\mathrm{j}\cdot 2k'\pi/N}&\overset{k'=k}{=}\sum_{n=0}^{(N-1)/2}\frac{[f(n)-f(n+1)](2n+1)}{\sum_{n=0}^{N-1}f(n)}\\&=\frac{f(0)+\sum_{n=1}^{(N-1)/2}2f(n)}{\sum_{n=0}^{N-1}f(n)}=1\end{aligned}\tag{2-69}$$

在 $k'\neq k$ 处的 Φ_s（jω）等于 0。apSW 幅频曲线同样通过 $\boldsymbol{H}$ 的频率采样点，采样点之间的幅度由式（2-64）插值函数确定。同法可求得 N（奇数）阶 apDW（前后窗均对称且相等），并注意到$\{R_{2n+1}(n)\}$构成非完备正交系，卷积窗谱可表示如下：

$$\Phi_D(j\omega) = \sum_{n=0}^{(N-1)/2} \frac{[f(n)-f(n+1)]}{N\sum_{n=0}^{N-1} f^2(n)} \left[\frac{\sin[(2n+1)\omega/2]}{\sin(\omega/2)}\right] \cdot \sum_{m=0}^{(N-1)/2} [f(m)-f(m+1)] \left[\frac{\sin[(2m+1)\omega/2]}{\sin(\omega/2)}\right] = \frac{\left\langle \sum_{n=0}^{(N-1)/2} \left\{ [f(n)-f(n+1)] \left[\frac{\sin[(2n+1)\omega/2]}{\sin(\omega/2)}\right] \right\} \right\rangle^2}{N\sum_{n=0}^{N-1} f^2(n)} \tag{2-70}$$

在频率采样点 $2k'\pi/N$ 处，有

$$\Phi_D[j(\omega-2k\pi/N)]\big|_{\omega=j\cdot 2k'\pi/N} \overset{k'=k}{=} \frac{\left[\sum_{n=0}^{N-1} f(n)\right]^2}{N\sum_{n=0}^{N-1} f^2(n)} < 1 \tag{2-71}$$

又由于 $\sin[(2n+1)\omega/2]/\sin(\omega/2)\leqslant(2n+1)$，所以在 $k'\neq k$ 处的频率点卷积窗谱值 $\Phi_D(j\omega)$ 均小于 1，此结论与定理 2 是一致的。MATLAB 实现的 apNW、apSW、apDW 卷积窗谱代码如下：

```
function apNSDSP(N)
% this function is to draw the none - single and dual windows all phase
% spectrum with rank N
clc,clf;
if nargin = =0
  N =8;
end
% two kinds of window
win(:,1) =boxcar(N);win(:,2) =triang(N);
% normalized coefficients
wconv =conv(win(:,2),win(:,2));
norms1 = [N,sum(win(:,2))];
norms2 = [1,wconv(N)/sum(win(:,2))];
% n angle frequency
w = -pi:2* pi/N^2:pi;nw =[0:N -1]'* w;
% fft
for n =1:2
```

```
    wins(:,:,n)=repmat(win(:,n),1,length(w));
    fjw(:,:,n)=sum(wins(:,:,n).* exp(-j* nw))/norms1(n);
    conjfjw(:,:,n)=sum(wins(:,:,n).* exp(j* nw))/norms2(n);
end
% build amplitudefreqeuency property
phiwn=fjw(:,:,1).* conjfjw(:,:,1)/N;
phiws=fjw(:,:,2).* conjfjw(:,:,1)/N;
phiwd=fjw(:,:,2).* conjfjw(:,:,2)/N;
% draw the results
plot(w,phiwn,'k-',w,phiws,'k--',w,phiwd,'k-.','Linewidth',3);
xlabel('\omega(rad/s)','FontSize',16);ylabel(' normalized amplitude
','FontSize',16);
grid on;legend('apNW','apSW','apDW');
title('spectrum interpolation','FontSize',16);
```

运行结果如图 2-34 所示。由图 2-34 可看出，apDW 在 $n=0$ 处幅值低于 1。由式（2-71）可知，幅值应该等于长度为 8 的三角窗所有元素的和的二次方除以元素的二次方和并除以 8，结果等于 0.7619。图 2-34 验证了这个结果的正确性。但平移的 Φ_{D}（jω）序列在 $n=0$ 处并不为零而将值提升，但始终不能达到 1。

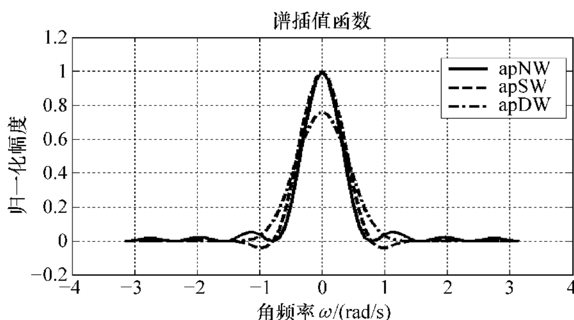


图 2-34　apNW、apSW 和 apDW 的谱插值函数图

定理 4：N 阶线性 ap 系统对卷积窗具有线性性。即如果 $\boldsymbol{C}=\alpha_0\boldsymbol{C}_0+\alpha_1\boldsymbol{C}_1+\cdots\alpha_{N-1}\boldsymbol{C}_{N-1}$，$y_i$ 是对应子卷积窗 $\boldsymbol{C}_i$ 的输出，则总输出 $y(n)=\alpha_0 y_0+\alpha_1 y_1+\cdots\alpha_{N-1}y_{N-1}$。

证明：对于 apDW 的卷积窗可以表示如下：

$$\boldsymbol{C}_n=\sum_{i=0}^{N-1}[f(i)\delta(n-i)]*b(-n)=\sum_{i=0}^{N-1}[f(i)b(i-n)]=\sum_{i=0}^{N-1}\boldsymbol{C}_n^i \tag{2-72}$$

式中

$$\boldsymbol{C}_n^i=f(i)b(i-n) \tag{2-73}$$

即卷积窗可以表示成 N 个子窗之和的形式，子窗的元素由式（2-73）构成。将子窗的表达形式代入式（2-45）得：

$$
\begin{aligned}
y(n) &= \sum_{k=-N+1}^{N-1}\left\{\left[\sum_{i=0}^{N-1} \boldsymbol{C}_k^i\right] h(k)x(n-k)\right\} = \sum_{i=0}^{N-1}\left\{\sum_{k=-N+1}^{N-1}\left[\boldsymbol{C}_k^i h(k)x(n-k)\right]\right\} \\
&= \sum_{i=0}^{N-1}\left\{\sum_{k=-N+1}^{N-1}\left[h'(k,\ i)x(n-k)\right]\right\} \\
&= \sum_{i=0}^{N-1}\left[h'(n,i) * x(n)\right] \\
&= \sum_{i=0}^{N-1} y_i,
\end{aligned} \tag{2-74}
$$

式中，$h'(n,\ i) = \boldsymbol{C}_k^i h(k)$。

由此可知，ap 系统卷积窗 $\boldsymbol{C}$ 分解成 N 个子窗时，总输出等于 N 个相应的子输出之和即 ap 系统对 $\boldsymbol{C}$ 具有线性可加性。由式（2-72）～式（2-74）推导过程易证明，ap 系统对 $\boldsymbol{C}$ 同样具有线性可乘性，所以 ap 系统对卷积窗 $\boldsymbol{C}$ 具有线性性。

设前后窗均为 4 阶三角窗的 ap 系统 $\boldsymbol{H}=[1\ 1\ 0\ 1]$，单位冲击响应序列 $\boldsymbol{h}=[0.25\ \ -0.25\ 0.25\ 0.75\ 0.25\ \ -0.25\ 0.25]$，分解得到子窗和子滤波器特性的 MATLAB 代码如下：

```
functionapDivW(N)
% this function is to decompose to get sub-convulation window
clc,close all;
if nargin = =0
  N =4;
end
% form the subconvulation window in Ci and sub hiap
winf =hanning(N);winb =hanning(N);convw =conv(winf,winb);
winf =winf/convw(N);H =[1 1 0 1]';htra =ifft(H);
hap =[htra(2:N);htra];Ci =zeros(2* N -1,1);
titlename ={ 'sub h1','sub h2','sub h3','sub h4' };
for i =1:N
    Ci(i:i +N -1) =winf(i)* winb
    hiap(:,i) =Ci.* hap;
    pout(:,:,i) =myfft(hiap(:,i),N);Ci =Ci -Ci;
    subplot(3,4,i),handle(i) =plot(pout(2,:,1),abs(pout(1,:,i)));
xlabel('\omega(rad/s)','FontSize',13);
ylabel('amplitude','FontSize',13);
    grid on;title(titlename{ i },'FontSize',13);
    subplot(3,4,i +4),
    handle(i +4) =plot(pout(2,:,1),angle(pout(1,:,i))/2/pi* 180);
    xlabel('\omega(rad/s)','FontSize',13);
    ylabel('phase','FontSize',13);grid on;
```

```
end
subplot(3,4,[9,10]),
handle(9)=plot(pout(2,:,1),sum(abs(pout(1,:,:)),3));
xlabel('\omega(rad/s)','FontSize',13);ylabel('amplitude','FontSize',13
);grid on;title('h1 +h2 +h3 +h4','FontSize',13);
subplot(3,4,[11,12]),handle(10)=plot(pout(2,:,1),
sum(angle(pout(1,:,:)),3));xlabel('\omega(rad/s)','FontSize',13);
ylabel('phase','FontSize',13);axis([0 pi -0.1 0.1]);
grid on;title('h1 +h2 +h3 +h4','FontSize',13 );
set(handle,'Color','k','LineWidth',3);
function pout =myfft(sin,N)
w =0:2* pi/N^3:pi;nw =[ -N +1:N -1]'* w;
hiap =repmat(sin,1,length(w));
pout(1,:) =sum(hiap.* exp( -j* nw));pout(2,:) =w;
```

运行结果如图 2-35 所示。

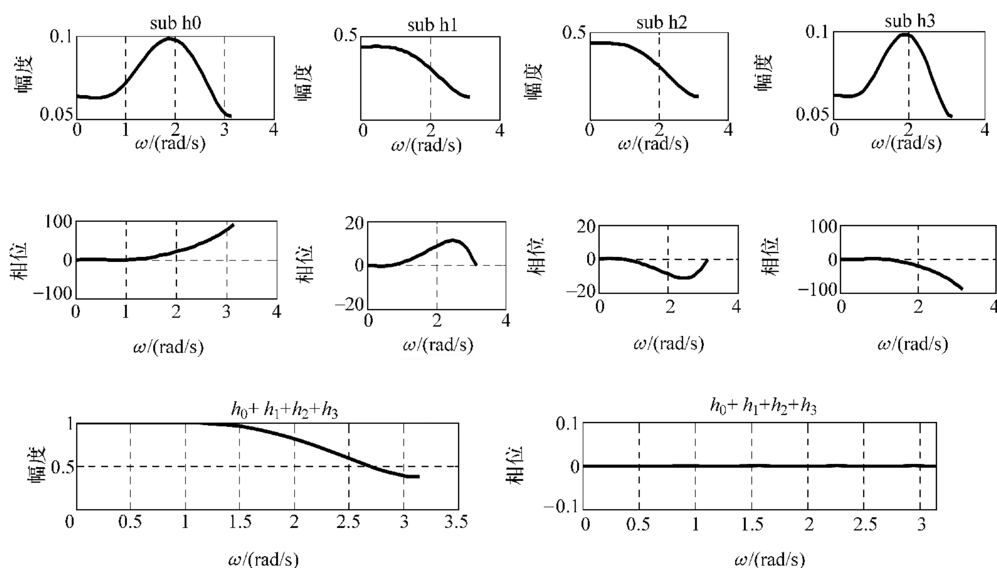


图 2-35　子滤波器幅频/相频及合成特性

分解得到的四个子窗分别为：$\boldsymbol{C}_n^{\ 0}$ = [0. 0637 0. 1667 0. 1667 0. 0637 0 0 0]，$\boldsymbol{C}_n^{\ 1}$ = [0 0. 0637 0. 1667 0. 1667 0. 0637 0 0]，$\boldsymbol{C}_n^{\ 2}$ = [0 0 0. 0637 0. 1667 0. 1667 0. 0637 0]，$\boldsymbol{C}_n^{\ 3}$ = [0 0 0 0. 0637 0. 1667 0. 1667 0. 0637]；对应的子滤波器的冲击响应为：$h'(n,0)$ = [0. 0159 －0. 0417 0. 0417 0. 0477 0 0 0]，$h'(n,1)$ = [0 －0. 0417 0. 1091 0. 3273 0. 0417 0 0]，$h'(n,2)$ = [0 0 0. 0417 0. 3273 0. 1091 －0. 0417 0]，$h'(n,3)$ = [0 0 0 0. 0477 0. 0417 －0. 04170. 0159]。由图 2-35 可看出子滤波器 0、3 及 1、2 有相同的幅频特性和互补的相位特性。由式（2-74）可知：

$$\begin{aligned} h'(N-1-n,N-1-i) &= C_{N-1-n}^{N-1-i}h(N-1-n) \\ &= f(N-1-i)b(n-i)h(N-1-n) \\ &= f(i)b(i-n)h(n) \\ &= h'(n,i) \end{aligned} \quad (2\text{-}75)$$

即第 i 个子滤波器是第 $N-1-i$ 个子滤波器的镜像，这与图 2-35 实验结果一致。具有零线性相位的 ap 分解得到的子滤波器都不再具有线性相位（N 为奇数时，仅有中间的子滤波器具有零线性相位），但合成后的滤波器保持严格零相位。

定理 5：N 阶线性 ap 系统对频率取样序列 $\boldsymbol{H}$ 具有线性性。即如果 $\boldsymbol{H}=\alpha_0\boldsymbol{H}_0+\alpha_1\boldsymbol{H}_1+\cdots\alpha_{N-1}\boldsymbol{H}_{N-1}$，$\boldsymbol{y}^m$ 是对应子卷积窗 $\boldsymbol{H}_{\mathrm{m}}$ 的输出，则总输出 $\boldsymbol{y}=\alpha_0\boldsymbol{y}^0+\alpha_1\boldsymbol{y}^1+\cdots\alpha_{N-1}\boldsymbol{y}^{N-1}$。

证明：设系统特性序列 $\boldsymbol{H}=\sum\boldsymbol{H}_m$，代入式（2-45）得：

$$\begin{aligned} y(n) &= \sum_{m=0}^{M-1}\left\{\frac{1}{N}\sum_{i=0}^{N-1}\left[s_{N-1-i}^{\mathrm{T}}\cdot\mathrm{diag}(\boldsymbol{B})\cdot\boldsymbol{W}^{*}\cdot\boldsymbol{H}_m\cdot\boldsymbol{W}\cdot(\mathrm{diag}(\boldsymbol{F})\cdot\boldsymbol{x}_{n+i})\right]\right\} \\ &= \sum_{m=0}^{M-1}y^m(n) \end{aligned} \quad (2\text{-}76)$$

式中，y^m（n）是对应子带 $\boldsymbol{H}_m$ 的输出，即 ap 系统对 $\boldsymbol{H}$ 具有可加性且可乘性明显，所以 ap 系统对频率抽样序列 $\boldsymbol{H}$ 具有线性性。因此，可用 ap 进行子带分解，而各个子带应包含信号的全部频率信息而不存在冗余部分即子带间应完全正交。这样，对于 N 阶 DFT 域最多可形成 $N/2+1$ 个（N 为偶数）或（$N+1$）/2 个（N 为奇数）个严格互补子带：

$$\begin{cases} \boldsymbol{H}=\bigcup_{m=0}^{N/2}\boldsymbol{H}_m \\ \bigcap_{m=0}^{N/2}\boldsymbol{H}_m=\boldsymbol{0} \end{cases} \quad (2\text{-}77)$$

子带 $\boldsymbol{H}_m$ 允许通过频率为 $2m\pi/N$ 的信号而抑制其他分量。如 4 阶全通 ap 系统 $\boldsymbol{H}=[1\ 1\ 1\ 1]$，最大分解得到的三个子带为：低通 $\boldsymbol{H}_0=[1\ 0\ 0\ 0]$、带通 $\boldsymbol{H}_1=[0\ 1\ 0\ 1]$ 和高通 $\boldsymbol{H}_2=[0\ 0\ 1\ 0]$，它们的幅频特性分别如图 2-36 所示。

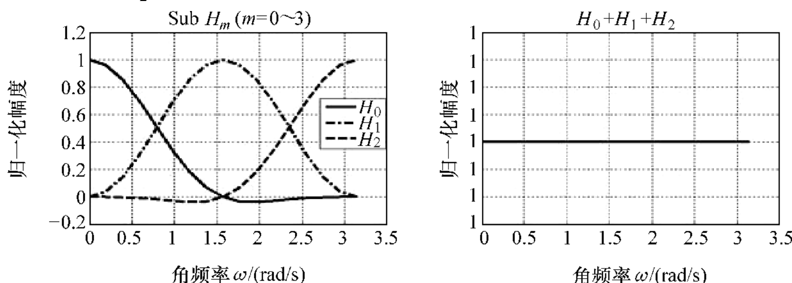


图 2-36　单频点子带分解图

实现的 MATLAB 代码如下：

```
function out=apSubH(N)
% this function performs to separate H to different sub Hm
clc,clf;
if nargin==0
  N=4;
end
% construct COS
w=0:pi/N^2:pi;cosw=w'*[0:N-1];
% constuct convulated window
wb=triang(N);wf=boxcar(N);
% normalized convulated window
wb=wb/sum(wb);C=conv(wb,wf);adh=alldiv(N,C,cosw);
% draw the amp-frequency curve of Hm
subplot(1,2,1),handle=plot(w,adh,'k','LineWidth',3),grid on;
xlabel('\omega(rad/s)','FontSize',16);ylabel('amplitude','FontSize',16);ti-
tle('sub Hm(m=0 ~ 3)','FontSize',16);
lstyle={'-','-.','--'};n=length(handle);
lgdname=['H0';'H1';'H2'];
for i=1:n
   set(handle(i),'LineStyle',lstyle{i});
end
hgd=legend(gca,lgdname);set(hgd,'FontSize',16);
subplot(1,2,2),plot(w,sum(adh,2),'k','LineWidth',3),grid on;
xlabel('\omega(rad/s)','FontSize',16);
ylabel('amplitude','FontSize',16);title('H0+H1+H2','FontSize',16);
function adh=alldiv(N,C,cosw)
% construct H
n=(mod(N,2)+N)/2+1-mod(N,2);
   Hm=zeros(N,n);Hm(1,1)=1;
for i=2:n
   Hm(i,i)=1;Hm(N+2-i,i)=1;
end
hm=ifft(Hm,N,1);
% generate Hg
cn=C(N:2*N-1);
for i=1:n
   adh(:,i)=sum(2*cos(cosw)*diag(cn.*hm(:,i)),2);
   adh(:,i) =adh(:,i) -C(N)*hm(1,i);
End
```

H 的分解方式在满足子带正交的条件下是任意的，如可以形成 2 倍频宽分解

方式即 $\boldsymbol{H}_k$ 宽度 2 倍于 $\boldsymbol{H}_{k-1}$。如 16 阶 ap 系统 2 倍频分解得到的 4 个子带 $\boldsymbol{H}_0$ =[1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]，$\boldsymbol{H}_1$ = [0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1]，$\boldsymbol{H}_2$ =[0 0 1 1 0 0 0 0 0 0 0 0 0 1 1 0]，$\boldsymbol{H}_3$ =[0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0]，幅频特性如图 2-37所示。

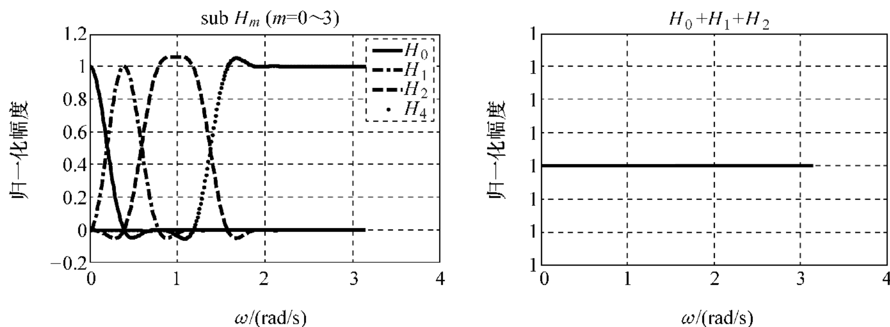


图 2-37　2 倍频子带分解图

实现代码（只列出 $\boldsymbol{H}_m$ 形成部分）如下：

```
functionsdh = semidiv(N,C,cosw)
% construct H
n =0;M =N;
while(M -2^n) >2^(n +1)
   M =M -2^n;n =n +1;
end
n =n +1;Hm =zeros(N,n);
Hm(1,1) =1;
for i =2:n -1
   Hm(i:2^(i -2) -1 +i,i) =1;
   Hm(N +3 -2^(i -2) -i:N +2 -i,i) =1;
end
Hm(:,n) =ones(N,1) -sum(Hm,2);hm =ifft(Hm,N,1);
% generate Hg
cn =C(N:2* N -1);
for i =1:n
   sdh(:,i) =sum(2* cos(cosw)* diag(cn.* hm(:,i)),2);
   sdh(:,i) =sdh(:,i)  -C(N)* hm(1,i);
end
```

为保持各信号分量相对幅值不变可将 $\boldsymbol{H}_m$ 均设置为 1，若要提高或降低某分量可直接将 $\boldsymbol{H}_m$ 中对应频点设定为变化倍数。对非线性相位 ap 系统的 $\boldsymbol{H}_m$ 取值可为复数，它同时对幅度和相位进行调节，如何根据应用需求来设置 $\boldsymbol{H}_m$ 是值得研究的方向。

定理 6：对输入信号进行 W-O 处理后的相关性降低，卷积双窗降低最大。

证明：由图 2-21 可知，ap 系统对输入信号的 W-O 处理可用图 2-38 示意说明。

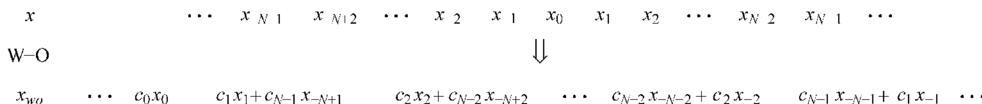


图 2-38　W-O 处理示意图

无限长序列 $\boldsymbol{x}$ 的自相关函数可取 $2N-1$ 长度来进行衡量，并假设相关性平稳（与时间起始点无关），可表示如下：

$$r_{xx}(m)=\frac{1}{2N-1}\sum_{i=-N+1}^{N-1}x(i)x^*(i+m) \tag{2-78}$$

则序列 x_{wo} 的自相关函数等于：

$$\begin{aligned}r_{x_{wo}x_{wo}}(m)&=\frac{1}{N}\sum_{i=0}^{N-1}x_{wo}(i)x_{wo}^*(i+m)\\&=\frac{1}{N}\sum_{i=0}^{N-1}(c_ix_i+c_{N-i}x_{-N+i})(c_ix_{i+m}^*+c_{N-i}x_{-N+i+m}^*)\\&=\frac{1}{N}\sum_{i=0}^{N-1}\left[c_i^2x_ix_{i+m}^*+c_ic_{N-i}(x_ix_{-N+i+m}^*+x_{-N+i}x_{i+m}^*)+c_{N-i}^2x_{-N+i}x_{-N+i+m}^*\right]\end{aligned} \tag{2-79}$$

式中，$c_N=0$。

设 $\boldsymbol{x}$ 为确定复指数信号序列，用最大幅值进行归一化后的元素 $x_i=AW^{\alpha i+\varphi}$，其中 $W=\mathrm{e}^{-\mathrm{j}2\pi/N}$，$\alpha$、$\varphi$ 为实变量。代入式（2-78）并令 $m=0$ 可得 $\boldsymbol{x}$ 最大自相关为

$$r_{xx}(0)=\frac{1}{2N-1}\sum_{i=-N+1}^{N-1}AW^{\alpha i+\varphi}AW^{-\alpha i-\varphi}=\frac{1}{2N-1}\sum_{i=-N+1}^{N-1}A^2=A^2 \tag{2-80}$$

将 x_i 表达式带入式（2-79）并令 $m=0$ 可得 x_{wo} 最大自相关：

$$\begin{aligned}r_{x_{wo}x_{wo}}(0)&=\frac{1}{N}\sum_{i=0}^{N-1}\left[c_i^2+2c_ic_{N-i}\cos(\alpha\cdot2\pi)+c_{N-i}^2\right]A^2\\&=\frac{1}{N}\sum_{i=0}^{N-1}\left[c_i^2+2c_ic_{N-i}\cos(\alpha\cdot2\pi)+c_{N-i}^2\right]r_{xx}(0)\\&\overset{\alpha=k+\Delta\alpha}{=}\frac{1}{N}\sum_{i=0}^{N-1}\left[c_i^2+2c_ic_{N-i}\cos(\Delta\alpha\cdot2\pi)+c_{N-i}^2\right]r_{xx}(0)\end{aligned} \tag{2-81}$$

对于 apNW、apSW 和 apDW 的卷积窗都满足 $c_i+c_{N-i}\leqslant1$ 且半带具有单调性。所以 $\boldsymbol{x}_{wo}$ 最大自相关满足

$$\frac{r_{x_{wo}x_{wo}}(0)}{r_{xx}(0)}<\frac{1}{N}\sum_{i=0}^{N-1}(c_i^2+c_{N-i}^2+2c_ic_{N-i})=\frac{1}{N}\sum_{i=0}^{N-1}(c_i+c_{N-i})^2\leqslant1 \tag{2-82}$$

在式（2-82）中 apDW 的卷积窗进行了第二次降低。所以，经过 W-O 处理后的信号相关性都会减弱，降低的最大情况出现在 apDW 时的卷积窗加权。

假设输入的复信号实部和虚部均是均值为 0 且方差为 1 的高斯白噪声，采用 N 阶进行 W-O 处理后的信号对比原信号自相关的 MATLAB 代码如下：

```
functionwoRxx(N)
if nargin = =0
  N =32;slen =N^2 +2* N -1;fontsize =16;
end
fwin(:,1) =boxcar(N);fwin(:,2) =triang(N);
bwin(:,1) =boxcar(N);bwin(:,2) =triang(N);
for i =1:2
    for j =1:2
        C(:,i +j -1) =conv(fwin(:,i),bwin(:,j));
    end
end
apNW =C(:,1)/N;apSW =C(:,2)/sum(fwin(:,2));
apDW =C(:,3)/C(N,3);
rs =rand(slen,1);is =rand(slen,1);sigin =rs +j* is;
[rxx,ryy,mval] =getrxy(N,rs,[apNW,apSW,apDW]);
tname ={ 'apNW','apSW','apDW' };
for k =1:3
    subplot(1,3,k),h1 =plot([rxx(:,k),ryy(:,k)],'k','LineWidth',3);
    set(h1(1),'LineStyle','-');set(h1(2),'LineStyle','-.');hold on;
    plot(ones(size(rxx,1),1)* mval(1,k),'k-','LineWidth',3);
    plot(ones(size(rxx,1),1)* mval(2,k),'k-.','LineWidth',3);
    axis tight;xlabel('n','FontSize',fontsize);
    ylabel('self correlation','FontSize',fontsize);
    h3 =legend('r_x_x','r_x_x_w_o');set(h3,'FontSize',fontsize);
    title(tname{ k },'FontSize',fontsize);grid on;
end
function[rxx,ryy,mval] =getrxy(N,rs,C)
for k =1:3
    for i =1:N
        xn =rs(i* N:(i +1)* N -1);
        rxx(i,k) =max(xcorr(xn,xn));
        xnwo =rs((i -1)* N +1:(i +1)* N -1).* C(:,k);
        yn =[xnwo(N);xnwo(1:N -1) +xnwo(N +1:2* N -1)];
        ryy(i,k) =max(xcorr(yn,yn));
    end
    mval(1,k) =mean(rxx(:,k));mval(2,k) =mean(ryy(:,k));
end
```

运行结果如图 2-39 所示。由图 2-39 可看出，经过 W-O 处理后的信号自相关性都减少了且 apDW 降低幅度最大，apNW 的降低幅度大于 apSW。又由式（2-81）可知，W-O 处理后信号自相关最大值是信号频移的函数，因此可用来对信号频率进行校正。

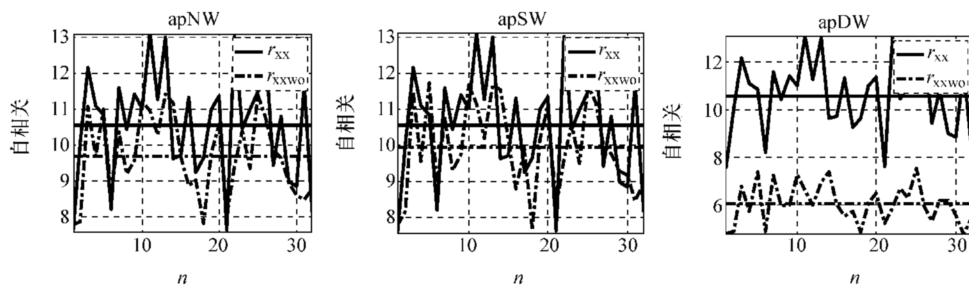


图 2-39　W-O 处理后信号自相关曲线

2.4　应用举例

由上节分析结果可知，ap 系统具有很多重要性质，可应用于多个领域如信号滤波，信号分析等方面。本节将举例说明 ap 的应用并讨论非整数倍信号频率抽样进行 DFT 而带来的频率栅栏效应及信号边界扩展所带来的误差问题。

2.4.1　基于 ap 的信号滤波

利用 ap 系统进行滤波的关键是频率采样序列 $\boldsymbol{H}$ 的设置，ap 系统具有严格零相位，所以只需对 $\boldsymbol{H}$ 设置就可以过滤出期望的信号而保持原有的相位信息。设输入信号频率成分集合递增排序后为 $\boldsymbol{F}_{\text{list}}=[f_0 f_1 \cdots f_{M-1}]$，为保证能够过滤出所有的信号分量需设定抽样频率 $f_s = kf_0(k \geqslant 2)$，则抽样得到的信号序列成分的数字角频率为 $\boldsymbol{\omega}_{\text{list}} = 2\pi \boldsymbol{F}_{\text{list}}/(kf_0)$，取其一阶差分的最小值为

$$\Delta\omega_{\min} = \min_{0 \leqslant k \leqslant M-1}\left\{\frac{2\pi}{k}\cdot\frac{f_k - f_{k-1}}{f_{M-1}}\right\} \quad \text{定义：} f_{-1}=0 \tag{2-83}$$

频率分辨率应小于式（2-83）的结果，即

$$2\pi/N' \leqslant \Delta\omega_{\min} \Rightarrow N' \geqslant \text{floor}\left[\frac{kf_{M-1}}{\min\limits_{0\leqslant k\leqslant M-1}\{f_k - f_{k-1}\}}\right]+1 \tag{2-84}$$

式中，floor[m]为不大于 m 的最大整数。一般取滤波器阶数 $N = rN'(r \geqslant 1)$ 以提升频率分辨能力，从而可保证在信号 $\Delta\omega_{\min}$ 内有 $r-1$ 个频率分辨率 ω_{res}。假设欲通过的 L 个频率成分为 $\boldsymbol{\omega}_{\text{p}} = [\omega_{\text{p}0}\omega_{\text{p}1}\cdots\omega_{\text{p}(L-1)}] \subset \boldsymbol{\omega}_{\text{list}}$。构造左右边通频点序列 $\boldsymbol{\omega}_{\text{lps}}/\boldsymbol{\omega}_{\text{rps}}$ 并映射成左右下标序列 $\boldsymbol{C}_{\text{l}}/\boldsymbol{C}_{\text{r}}$：

$$\boldsymbol{\omega}_{\mathrm{lps}}=\boldsymbol{\omega}_{\mathrm{p}}-\omega_{\mathrm{res}}=[\omega_{\mathrm{p}0}-\omega_{\mathrm{res}}\quad \omega_{\mathrm{p}1}-\omega_{\mathrm{res}}\quad \cdots\quad \omega_{\mathrm{p}(L-1)}-\omega_{\mathrm{res}}]\xrightarrow{\text{映射 1}}\boldsymbol{C}_{\mathrm{l}}=[c_{\mathrm{l}0}\quad c_{\mathrm{l}1}\quad \cdots\quad c_{\mathrm{l}(L-1)}]$$
$$\boldsymbol{\omega}_{\mathrm{rps}}=\boldsymbol{\omega}_{\mathrm{p}}+\omega_{\mathrm{res}}=[\omega_{\mathrm{p}0}+\omega_{\mathrm{res}}\quad \omega_{\mathrm{p}1}+\omega_{\mathrm{res}}\quad \cdots\quad \omega_{\mathrm{p}(L-1)}+\omega_{\mathrm{res}}]\xrightarrow{\text{映射 2}}\boldsymbol{C}_{\mathrm{r}}=[c_{\mathrm{r}0}\quad c_{\mathrm{r}1}\quad \cdots\quad c_{\mathrm{r}(L-1)}]$$
$$(2\text{-}85)$$

映射方式如下：

$$\text{映射}:c_{\mathrm{l}i}=\frac{\omega_{\mathrm{p}i}-\omega_{\mathrm{res}}}{\omega_{\mathrm{res}}},c_{\mathrm{l}i}=\frac{\omega_{\mathrm{p}i}+\omega_{\mathrm{res}}}{\omega_{\mathrm{res}}},\ i=0,1,\cdots,L-1 \tag{2-86}$$

将 ap 系统特性中坐标为（$\boldsymbol{C}_{\mathrm{l}}$，$\boldsymbol{C}_{\mathrm{r}}$）的部分置为 1，就得到了期望的频率抽样序列 $\boldsymbol{H}$，按照 ap 系统对综合信号进行频域处理就得到给定频率的信号分量。

对于给定的 N，某频率成分信号在一个周期内抽样点数可能不是整数，因此，即使时域幅度与其他信号相同，但在 DFT 后频域中会出现旁瓣且幅值会各不相同。假设某频率在单个周期内抽样点数为 M 且 $N=uM$（u 为整数），做 N 点 DFT 得：

$$\begin{aligned}X(k)&=\sum_{n=0}^{N-1}\left(\cos\frac{2\pi}{N}un\right)W_N^{kn}=\sum_{n=0}^{N-1}\left(\cos\frac{2\pi}{u\cdot M}un\right)W_N^{kn}\\&=\sum_{n=0}^{wM-1}\left(\cos\frac{2\pi}{M}n\right)\mathrm{e}^{-\mathrm{j}\frac{2\pi}{M}\frac{k}{u}n}\end{aligned} \tag{2-87}$$

由正交性可知，式（2-87）求和元素中与信号相同或互补的频率成分保留，其余部分求和后均为 0，故有（只给出相同成分，互补成分 $N-u$ 可作类似推导）

$$X(k)=\sum_{n=0}^{uM-1}(\cos^2\frac{2\pi}{M}n)\delta(u)=\frac{1}{2}\sum_{n=0}^{uM-1}(1+\cos\frac{4\pi}{M}n)\delta(u)=\frac{N}{2}\delta(u) \tag{2-88}$$

式（2-88）表明，单位幅度且频率为 $2u\pi/N$（u 为整数）余弦信号进行长度为 N（$N=Mu$）的 DFT 变换，其谱是幅值为 $N/2$ 位于 u 和 $N-u$ 处冲击。由于没有旁瓣，该信号可被无失真滤出。当 N 不是 M 的整数倍时[即 $N=uM+s$（u 为整数，$0<s<M$）]，DFT 变换结果为

$$\begin{aligned}X(k)&=\sum_{n=0}^{N-1}\left\{\cos\frac{2\pi}{N}\left(u+\frac{s}{M}\right)n\right\}\mathrm{e}^{-\mathrm{j}\frac{2\pi}{N}kn}\\&=\frac{1}{2}\frac{\sin\left(u+\frac{s}{M}-k\right)\pi}{\sin\left(u+\frac{s}{M}-k\right)\frac{\pi}{N}}\mathrm{e}^{\mathrm{j}\frac{N-1}{N}\left(w+\frac{s}{M}-k\right)\pi}+\frac{1}{2}\frac{\sin\left(u+\frac{s}{M}+k\right)\pi}{\sin\left(u+\frac{s}{M}+k\right)\frac{\pi}{N}}\mathrm{e}^{-\mathrm{j}\frac{N-1}{N}\left(w+\frac{s}{M}+k\right)\pi}\end{aligned}$$
$$(2\text{-}89)$$

由式（2-89）可看出，当 s 接近 0 或者 M 时，频谱主瓣在 u 和 $N-u$ 处；当 s 接近 $M/2$ 时，频谱主瓣在 $u+1$ 和 $N-u-1$ 处。此外，由于非 0 量 s 的存在而产生了 $N-2$ 个旁瓣分布在主瓣两边，幅值随着与主瓣的距离增加而渐弱，如图 2-40所示。当 s 为 $0\sim M$ 时，主瓣幅值变化区间为 $0.6N\sim N$。由图 2-40 可看出，旁瓣偏离均方值在 $M/2$ 处达到最大值。

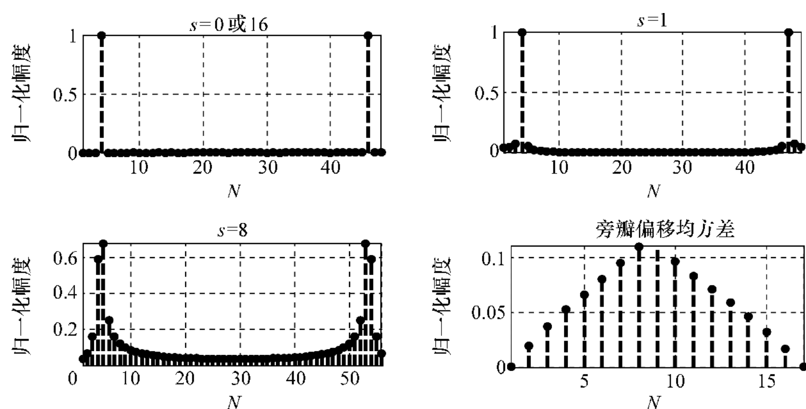


图 2-40　不同 s 时旁瓣幅值及偏离方差

在上面分析的基础上，我们通过一个实例来验证了 ap 滤波的优良性能。设复合信号由下式三种成分构成：

$$f(t)=1.2\cos(100t+0.9\pi)+0.8\cos(1500t+0.2\pi)+2.3\cos(400t+0.7\pi) \tag{2-90}$$

则频集 $\boldsymbol{F}_{\text{list}}=[100\ 400\ 1500]$，并取 $k=4$ 和 $r=1$，抽样频率则为 $f_s=4\times1500\text{Hz}=6000\text{Hz}$，滤波器阶数等于 $6000/100+1=61$，取 $N=64$ 以便进行 FFT 运算。复合信号和 $r=1$ 时频谱分析结果如图 2-41 所示。由图 2-41 中分别按照 ap 系统和传统方法恢复信号结果对比可看出，ap 结果明显克服了传统方法分段带来的间断问题。

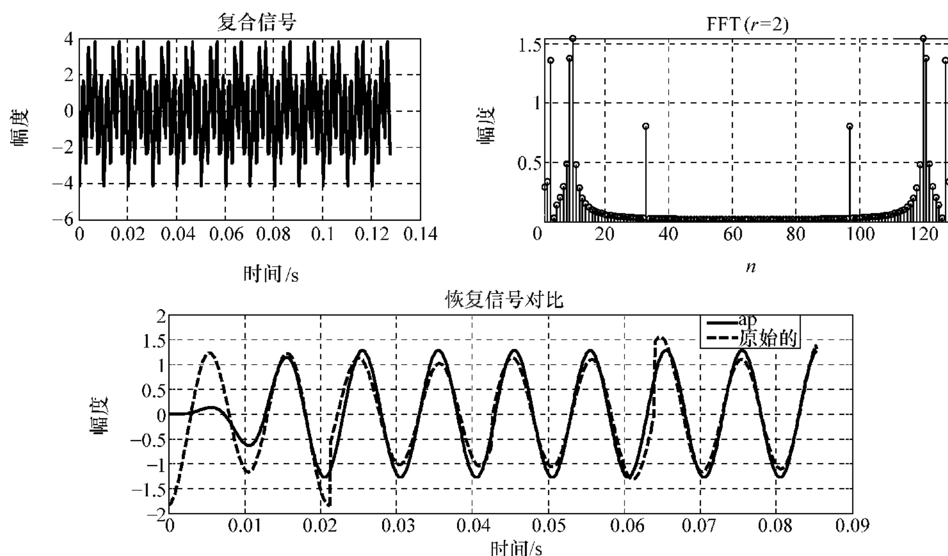


图 2-41　复合信号及 FFT 谱和恢复信号对比

2.4.2 边界拓展

需要注意是 64 阶 ap 延迟为 63 个单位时间，所以 $r=1$ 时正确结果的开始时刻 $t_d=63/6000\text{Hz}=0.0105\text{s}$，图 2-40 的结果也证实了此结论。即对于长度为 L 的信号经 N 阶 ap 系统处理后，长度将变为 $L-2N+2$。为了减小初始 $N-1$ 个结果的失真并保持信号实际长度，需要对输入信号作边界扩展的预处理。边界扩展应在输入信号的两端进行，通常有边界重复、周期扩展和对称延拓三种方式，示意图如图 2-42 所示。

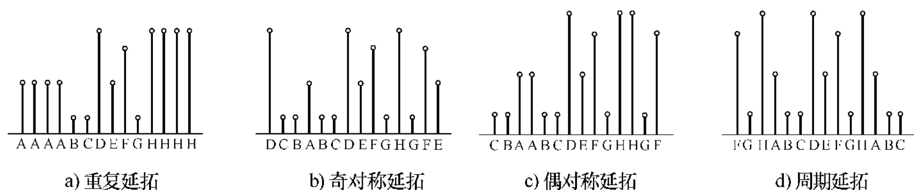


图 2-42 四种拓展方式示意图

分别按四种方式对含三种频率成分的复合采样信号进行拓展，其中信号成分 s_0 的初相为 $\pi/7$，复合信号构成如下：

$$\boldsymbol{S}=3\cos\left(\frac{2n\pi}{64}+\frac{\pi}{7}\right)+\cos\left(\frac{2n\pi}{20}\right)+2.3\cos\left(\frac{2n\pi}{4}\right) \tag{2-91}$$

信号长度取 256。用 64 阶 ap 对四种拓展信号过滤出 s_0 及误差对比如图 2-43 所示。

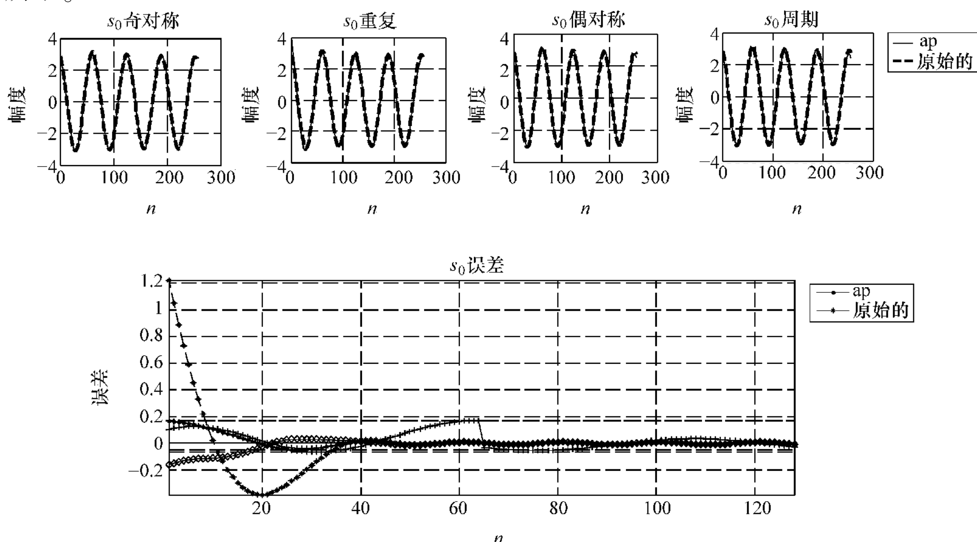


图 2-43 ap 四种延拓与传统法恢复信号误差比较

由图 2-43 可看出，四种拓展方式在经历 64 个采样周期后误差均小于传统方法，其中以奇对称误差最小。图 2-43 再次证实，ap 较好地克服了传统分段引起的“跳跃”问题。采用奇对称延拓 ap 恢复与传统方法得到的 3 种分量误差对比结果如图 2-44 所示。

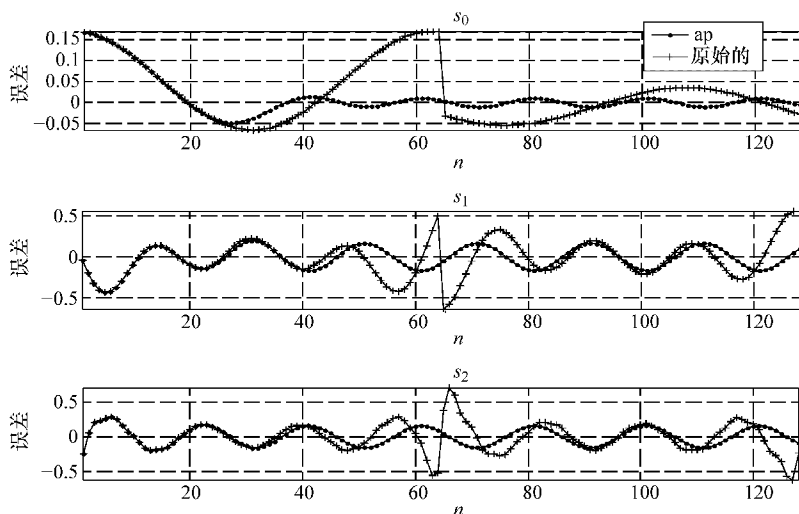


图 2-44　ap 奇延拓与传统法恢复 $s_0 \sim s_2$ 误差比较

实现上面实验的 MATLAB 代码如下：

```
function out = apexpand(frelist)
siglist = [3.0 1.0 2.3;1/64 1/20 1/4;1/7 0 0];
[sigout,NN,H] = concoct(siglist);
signals = sexpand(sigout,NN,'o');
apout = apflt(H,signals);oriout = orflt(H,signals);
[L N M] = size(apout);sname = { 's0 ','s1 ','s2 ' };
tname = { ' odd',' reptition',' even',' periodic','original' };
tagtp = { '.','* ',' + ','d','s' };
for i =1:M
   for j =1:N
      subplot(M,N,N* (i -1) +j),h =plot(apout(:,j,i),'k','LineWidth',1);
      hold on,plot(oriout(:,j,i),'k - - ','LineWidth',3);grid on;
      xlabel('n','FontSize',16);ylabel('amplitude','FontSize',16);
      title(strcat(sname{ i },tname{ j }),'FontSize',16);
   end
end
h = legend('ap','original');set(h,'FontSize',16);
coss = [0:L -1]'* siglist(2,:)* 2* pi;
```

```
amps=repmat(siglist(1,:),L,1);
phas=repmat(siglist(3,:),L,1)* pi;
orisig=amps.* cos(coss+phas);
for n=1:M
    orisigsex(:,:,n)=repmat(orisig(:,n),1,N);
end
aperrs=apout-orisigsex;orerrs=oriout-orisigsex;figure
for i=1:M
    subplot(M,1,i),
    for j=1:N
        h=plot(aperrs(1:2* NN,j,i),'k','LineWidth',1);
        set(h,'Marker',tagtp{ j },'MarkerSize',6);hold on;
    end
    h=plot(orerrs(1:2* NN,1,i),'k','LineWidth',1);
    set(h,'Marker',tagtp{ 3 },'MarkerSize',6);
    xlabel('n','FontSize',16);ylabel('errors','FontSize',16);
    title(strcat(sname{ i },' errors'),'FontSize',16);axis tight;
    h=legend('ap','original');set(h,'FontSize',16);grid on;
end
out=H1;
function apout=apflt(H,input)
N=size(H,1);sigsize=size(input);
wf=chebwin(N);wb=boxcar(N);
wconv=conv(wf,wb);wf=wf/sum(wf);
wfs=repmat(wf,1,sigsize(2));wbs=repmat(wb,1,sigsize(2));
nH=size(H,2);
for n=1:nH
    Hs=repmat(H(:,n),1,sigsize(2));
    d=zeros(N,sigsize(2));
    for i=1 :sigsize(1)-N+1
      infft=fft(wfs.* input(i:i+N-1,:));
      outfft=Hs.* infft;out=ifft(outfft).* wbs;
      apoutput(i,:,n)=d(1,:)+out(1,:);
      for j=1:N-2
          d(j,:)=d(j+1,:)+out(j+1,:);
      end
      d(N-1,:)=out(N,:);
    end
end
apout=apoutput(N:sigsize(1)-N+1,:,:);
function oriout=orflt(H,signal)
```

```
N=size(H,1);nH=size(H,2);
flen=(size(signal,1)-2*N+2)/N;
for j=1:nH
    Hs=repmat(H(:,j),1,size(signal,2));
    for i=1 :flen
      af=fft(signal(i*N:(i+1)*N-1,:)).*Hs;
      oriout((i-1)*N+1:i*N,:,j)=ifft(af);
    end
end
function[signal,N,H]=concoct(siglist)
N=64;slen=256;M=3;H=zeros(N,M);
nones=floor(N*siglist(2,[1:M]));
H(1,:)=1;
for n=1:M
    H(2:1+nones(n),n)=1;H(N+1-nones(n):N,n)=1;
end
H(:,2)=H(:,2)-H(:,1);
H(:,3)=H(:,3)-H(:,2)-H(:,1);
% to make out the compound frquency signal with fretable...
atf=repmat(siglist(1,:),slen,1);ptf=repmat(siglist(3,:),slen,1);
costf=cos([0:slen-1]'*siglist(2,:)*2*pi+ptf*pi);
signal=sum(costf.*atf,2);
function[sigout]=sexpand(signal,N,expt)
switch expt
case 'r'
    sigout=[ones(N-1,1)*signal(1);signal;ones(N-1,1)*signal(end)];
case 'e'
    sigout=[fliplr(signal(1:N-1));signal;fliplr(signal(end -N+2:end))];
case 'o'
    sigout=[fliplr(signal(2:N));signal;fliplr(signal(end-N+1:end-1))];
case 'p'
    sigout=[signal(end-N+1:end-1);signal;signal(2:N)];
Otherwise
sigout(:,1)=[fliplr(signal(2:N));signal;fliplr(signal(end-N+1:end-1))];
sigout(:,2)=[ones(N-1,1)*signal(1);signal;ones(N-1,1)*signal(end)];
sigout(:,3)=[fliplr(signal(1:N-1));signal;fliplr(signal(end -N +2:end))];
sigout(:,4)=[signal(end-N+1:end-1);signal;signal(2:N)];
end
```

2.4.3　噪声多带分解

以基于 64 阶 ap 系统对噪声的多频带分析作为本章的结束。设信号源是长度

为 200 的零均值高斯白噪声且方差约为 1。利用 64 阶 ap 进行二倍频分析处理，形成的 6 个子带对应的信号分量如图 2-45 所示。

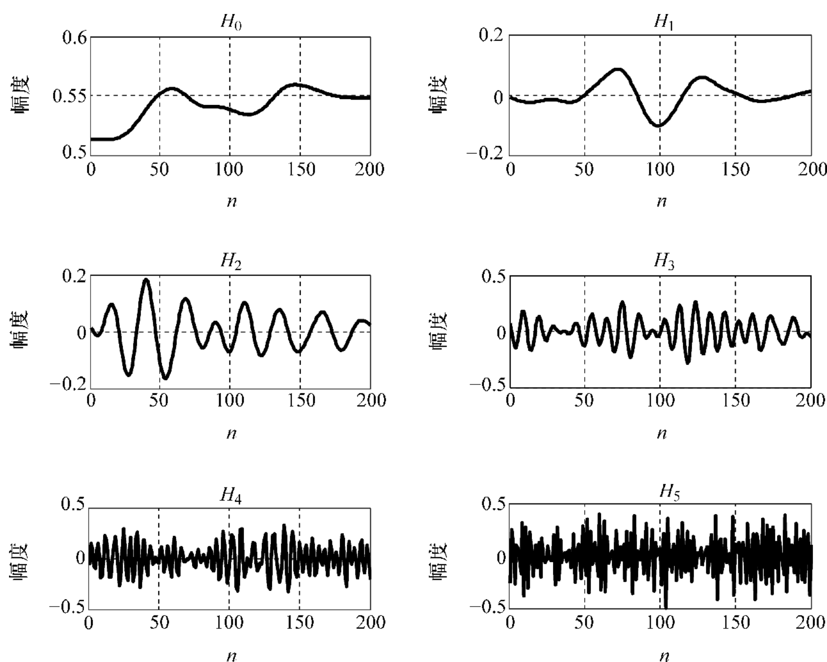


图 2-45　基于 ap 的二倍频信号分解

基于以上实验用到的子带分解函数 semidiv、延拓函数 sexpand 和 ap 处理函数 apflt 实现的 MATLAB 代码如下：

```
function out =apsubnoise(N)
% this function performs to expand the signal in different ways
clc,close all;
if nargin = =0
  N =64;
end
Hm =semidiv(N);signal =rand(200,1);
sigext =sexpand(signal,N,'o')
apout =apflt(Hm,sigext);
n =size(Hm,2);
for i =1:n
   subplot(n/2,2,i),plot(apout(:,1,i),'k','LineWidth',3);
   xlabel('n','FontSize',16),ylabel('amp','FontSize',16);
   title(strcat('H',num2str(i -1)),'FontSize',16);
   grid on;
end
```

2.5　小结

本章首先给出 DFT 域 ap 系统实现，经推导得到与其等效的时域 FIR 结构（ap1）和频域结构（ap2）并进而给出带窗的 ap 系统，还对 apNW、apSW 和 apDW 的性能进行了对比，得出了 apDW 具有最优的衰减特性的结论。其次，提出 ap 的线性相位特性、对 $\boldsymbol{H}$ 和 $\boldsymbol{C}$ 的线性性，以及满足频率抽样定理等重要性质，并结合实验给出了证明。最后，利用 ap 的性质将其运用于信号滤波和对噪声的多子带分析两方面，得到了一些重要的结论，同时也提出了一些有待解决和改进的问题。

第 3 章 基于全相位的窗函数设计

3.1 滤波器设计中的窗函数

数据窗在 FIR 滤波器窗函数设计中起着重要作用，它的性能好坏直接影响着滤波器的过渡带宽度和衰减大小。在 ap 系统设计中的卷积窗 $\boldsymbol{C}$ 对滤波器的过渡带宽度、通带及阻带内的波纹数也有很大影响。在数字信号处理的其他方面如谱估计等，窗函数同样起着重要作用。

3.1.1 窗的作用

对信号进行各种处理时，由于受到处理时间和计算机存储容量的限制，实际被处理的只能是有限长度的样本，即原始信号必然要被截短，相当于用矩形窗对信号进行调制，从而导致泄漏误差。设 $\boldsymbol{h}_d$是理想低通滤波器的单位冲击响应序列，$\boldsymbol{h}$ 是从长度为 N 的矩形窗口 $\boldsymbol{R}_N$看到的一段序列（即 $\boldsymbol{h}_d=\boldsymbol{h}\cdot\boldsymbol{R}_N$），则滤波器频谱为

$$H(e^{j\omega})=\frac{1}{2\pi}\int_{-\pi}^{\pi}H_d(e^{j\theta})R_N(e^{j(\omega-\theta)})d\theta \tag{3-1}$$

式中，$H_d(e^{j\omega})$ 和 $R_d(e^{j\omega})$ 分别是 $\boldsymbol{h}_d$和 $\boldsymbol{R}_N$的傅里叶变换。矩形窗谱函数等于：

$$\begin{aligned}R_N(e^{j\omega}) &= \sum_{n=0}^{N-1}R_N(n)e^{-jn\omega} = \sum_{n=0}^{N-1}e^{-jn\omega}\\ &= e^{-j\frac{N-1}{2}\omega}\frac{\sin(N\omega/2)}{\sin(\omega/2)} = R_{Ng}(\omega)e^{-j\alpha\omega}\end{aligned} \tag{3-2}$$

$$R_{Ng}(\omega)=\sin(N\omega/2)/\sin(\omega/2),\alpha=(N-1)/2$$

线性理想的低通滤波器 $H_d(e^{j\omega})=H_{dg}(\omega)e^{-j\alpha\omega}$，幅频特性 $H_{dg}(\omega)$ 满足：

$$H_{dg}(\omega)=\begin{cases}1 & |\omega|\leqslant\omega_c\\ 0 & \omega_c<|\omega_c|\leqslant\pi\end{cases} \tag{3-3}$$

把式（3-2）和式（3-3）代入到式（3-1）得到：

$$
\begin{aligned}
H(\mathrm{e}^{\mathrm{j}\omega}) &= \frac{1}{2\pi}\int_{-\pi}^{\pi} H_{\mathrm{dg}}(\theta)\mathrm{e}^{-\mathrm{j}\alpha\theta}R_{\mathrm{Ng}}(\omega-\theta)\mathrm{e}^{-\mathrm{j}\alpha(\omega-\theta)}\mathrm{d}\theta \\
&= \mathrm{e}^{-\mathrm{j}\alpha\omega}\frac{1}{2\pi}\int_{-\pi}^{\pi} H_{\mathrm{dg}}(\theta)R_{\mathrm{Ng}}(\omega-\theta)\mathrm{d}\theta = H_{\mathrm{g}}(\omega)\mathrm{e}^{-\mathrm{j}\alpha\omega} \qquad (3\text{-}4) \\
H_{\mathrm{g}}(\omega) &= \frac{1}{2\pi}\int_{-\pi}^{\pi} H_{\mathrm{dg}}(\theta)R_{\mathrm{Ng}}(\omega-\theta)\mathrm{d}\theta = H_{\mathrm{dg}}(\omega) * R_{\mathrm{Ng}}(\omega)
\end{aligned}
$$

式（3-4）说明滤波器幅频特性等于理想低通滤波器幅频特性 $H_{\mathrm{dg}}(\omega)$ 与矩形窗幅频特性 $R_{\mathrm{Ng}}(\omega)$ 的卷积，形成过程可简单地由图 3-1 来描述。

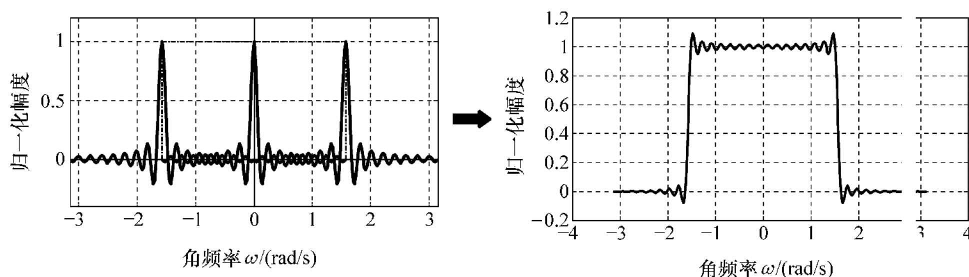


图 3-1　卷积过程构造的低通滤波器

实现上述结果的 MATLAB 代码如下：

```
function sic2fir(N)
% this function is to plot descripe the convoluation process of rectangle
% and sin(x)/x function to construct the low pass filter
clc,clf,close all;

if nargin = =0
   N =64;side =pi/2;dev =0;
   xlimit =2* pi/2;ylow = -0.4;yhigh =1.2;
end
w = -side:side/N:side;w(w = =0) =eps;
snw =sin(N* w/2);sw =sin(w/2);amp =snw./sw/N;
plot(w,amp,'k -',w -side,amp,'k -',w +side,amp,'k -','LineWidth',2);
hold on;grid on;xlabel('\omega(rad/s)');ylabel('amp');
R =ones(size(w));lside =0:0.1:1;xl =ones(size(lside))* (-side);
xr =ones(size(lside))* (side);
plot(w,R,'k -.',xl,lside,'k -.',xr,lside,'k -.','LineWidth',2);
xcor = -xlimit:xlimit/N:xlimit;xcory =zeros(size(xcor));
ycor =ylow:(yhigh -ylow)/N:yhigh;ycorx =zeros(size(ycor));
plot(xcor,xcory,'k',ycorx,ycor,'k','LineWidth',2);
```

```
axis([-xlimit xlimit ylow yhigh]);
wext=-3*side:side/N:3*side;wext(wext==0)=eps;
snw=sin(N*wext/2);sw=sin(wext/2);y=snw./sw/N;
    wlen=length(w);fv=zeros(1,4*N);

for n=0:4*N-1
    yv=y(end-wlen+1-n:end-n);
    fv(n+1)=trapz(yv);
end
figure(2),plot([-2*side:side/N:2*side-side/N],fv/4,'k','LineWidth',3);
grid on;xlabel('\omega(rad/s)');ylabel('amp');
```

当 $\omega=0$ 时，H（0）为图 3-1 中当 $R_{Ng}(\omega)$ 处于 $H_{dg}(\omega)$ 中央位置时两者乘积的积分，相当于对 R_{Ng}（ω）在［$-\omega_c$，ω_c］之间波形积分；当 $\omega_c\geqslant 2\pi/N$ 时，此结果近似在［$-\pi$，π］之间波形积分，将 $H(0)$ 值归一化到 1；当 ω 等于截止频率 ω_c 时即图 3-1 左右两个 R_{Ng}（ω），积分近似为一半波形积分时，相对 H（0）归一化后的值为 0.5；当 $\omega=\omega_c+2\pi/N$ 时，$R_{Ng}(\omega)$ 主瓣完全移到积分区间 $H_{dg}(\omega)$ 外面，最大负峰完全在区间［$-\omega_c$，ω_c］中，因此 $H_g(\omega)$ 在该点形成最大的负峰；当 $\omega=\omega_c-2\pi/N$ 时，$H_g(\omega)$ 产生最大正峰。最大正峰与最大负峰对应的频率相距 $4\pi/N$。通过以上的分析，对 $\boldsymbol{h}_d$ 加窗处理后滤波器幅值特性 H_g（ω）与原理想特性 $H_{dg}(\omega)$ 的差别主要有以下几点：

1）ω_c 附近产生宽度约为 $4\pi/N$ 即主瓣宽度的过渡带。

2）通带内增加了波动，在 $\omega_c-2\pi/N$ 处出现过冲为 0.089 的最大正峰。

3）阻带内产生了余振，在 $\omega_c+2\pi/N$ 处出现过冲为 -0.090 的最大负峰。

4）$H_{dg}(\omega)$ 波动愈快（N 加大时），$H_g(\omega)$ 通带和阻带内波动愈快；

5）$H_{dg}(\omega)$ 旁瓣大小直接影响 $H_g(\omega)$ 波动大小。

图 3-2 是当 N 分别取 4、8、16、32 时 $H_{dg}(\omega)$ 和 $H_g(\omega)$ 的波纹变化。

以上五点就是所谓的吉布斯（Gibbs）效应，它直接影响滤波器性能。通带内波动影响滤波器通带平稳性，阻带波动影响阻带最小衰减，同时滤波器一般都要求过渡带越窄越好。增加 N 可有效地控制过渡带宽度，但并不是减少吉布斯效应的有效方法。通过以下实验可证实此结论，研究当 $H_{dg}(\omega)$ 的主瓣右零点处于矩形窗右边界点作为右移动起始位置，以频率分辨率 $2\pi/N$ 为步长，则 H_{dg}（ω）主瓣及单边各个旁瓣的谱面积（有负值）之和即为最大正峰、过渡点、最大负峰及各个阻带波纹的幅度。同时，还可统计阶数 N 变化时最大正峰值和主瓣能量占比的变化规律。

图 3-3 中，当阶数 N 由 2^4 增大到 2^{16} 时，由于旁瓣数量的增加致主瓣能量占比逐渐降低，最大正峰的过冲程度也随着 N 的增大而逐渐减小，但阶数增大至 2^{10} 以后，过冲程度基本没有再改善。实现上面结果的 MATLAB 代码如下：

图 3-2　不同阶数引起的波纹数变化

图 3-3　主瓣及旁瓣能量分布和阶数 N 对最大正峰影响

```
function sicLobes(N)
% this function is to draw sin(Nw/2)/sin(w/2)with different N and
% calculate the areas covered by different side lobes

if nargin = =0
   N =2^6;M =N* 4;
```

```
    wlimit = pi;nlow = 4;nhigh = 16;
end
[sumya,norya,nn] =  lobestat(N,M,wlimit);

[ha,hl1,hl2] = plotyy([1:nn],sumya,[1:nn],norya,'plot','stem');
set([hl1,hl2],'Color','k','Linewidth',2);
set(hl1,'LineStyle','--','Marker','.','MarkerSize',20);
set(ha(1),'FontSize',16);set(ha(2),'FontSize',16);
hold on;grid on;xlabel('nth lobes');ylabel('normalized area');
legend('sum of sliding side lobes','spectrum of side lobes');
title('spectrum of all side lobes');
for k = 1:nn
    text(k,sumya(k),num2str(sumya(k)));
end
for n = nlow:nhigh
    N = 2^n;M = N* 4;
    [sumy2N(n - nlow +1),rm2N(n - nlow +1)] = lobestat(N,M,wlimit);
end
figure,[ah,hl1,hl2] = plotyy([nlow:nhigh],sumy2N,[nlow:nhigh],rm2N);
set([hl1,hl2],'Color','k','Linewidth',2);
set(hl1,'LineStyle','--','Marker','.','MarkerSize',20);
set(hl2,'LineStyle','-.','Marker','o','MarkerSize',8);
set(ah(1),'FontSize',16,'XMinorTick','on');
set(ah(1),'FontSize',16,'YMinorTick','on');
set(ah(2),'FontSize',16,'YMinorTick','on');
xlabel('order N');ylabel(ah(1),'overshoot');
ylabel(ah(2),'main lobe spectrum percentage');grid on;
legend('max positive overshoot','main lobe percentage');
title('overshoot and main lobe changing with order N');

function varargout = lobestat(N,M,wlimit)
w = -wlimit:2* wlimit/M:wlimit;w(w ==0) = eps;
snw = sin(N* w/2);sw = sin(w/2);amp = snw./sw/N;
nn = N/2 +1;
for k = 1:nn
    nstep = M/N;
    wstart = (k -2)* 2* pi/N + wlimit;
    nstart = k -2 + N* wlimit/pi/2;nend = nstart +1;
    y = amp(nstart* nstep +1:nend* nstep +1);
    yarea(k) = trapz(y);
end
```

```
norya = yarea/sum([yarea([1,2]),2* yarea(3:end)]);
mlober = (yarea(1) + yarea(2))/sum(abs([yarea([1,2]),2* yarea(3:end)]));
for k = 1:nn
    sumya(k) = sum(norya(k:nn));
end
ifnargout = =2
    varargout{1} = sumya(1);varargout{2} = mlober;
else
    varargout{1} = sumya;varargout{2} = norya;varargout{3} = nn;
end
```

减小通带波动及增大阻带衰减只能从窗函数形状上解决，一般要求窗函数谱主瓣尽量包含较多的能量，相应旁瓣幅度就会相对地减小，从而使通带和阻带波动减小且加大阻带衰减，但会以过渡带加宽为代价。在各种窗函数中，阻带衰减大小和带宽宽度是相互矛盾的，选取和设计中应按实际要求侧重一方。

3.1.2　传统窗特性

对窗向量 $\boldsymbol{w}$ 总的要求是希望其频谱中主瓣尽量窄，边瓣幅度尽量小，即频域能量主要集中在主瓣内。此外，$\boldsymbol{w}$ 还应满足下列要求以便可定量地比较各窗性能：

➢ 非负实偶且从对称中心向两边非递增；

➢ 保证滤波器的通带增益为 1，即

$$\boldsymbol{w}(0) = \frac{1}{2\pi}\int_{-\pi}^{\pi} W(\mathrm{e}^{\mathrm{j}\omega})\mathrm{d}\omega = 1$$

➢ 为保证滤波器相位特性不因加窗而改变，一般要求 $\boldsymbol{w}$ 恒为正。

定义如下三个衡量窗函数性能指标的参数：

1）3dB 带宽 B，它是主瓣归一化幅度（$20\log_{10}\left|\frac{W(\mathrm{e}^{\mathrm{j}\omega})}{W(0)}\right|$）下降到 -3dB 时的带宽。当数据长度为 N 时，最大可能的频率分辨率是 $\Delta\omega = 2\pi/N$，则 B 的单位可以是 $\Delta\omega$。

2）最大旁瓣峰值 A（dB），A 越小表明由旁瓣引起的振荡幅度越小。

3）旁瓣峰值渐进衰减速度 D（dB/oct，oct 表示倍频程）。

显然，理想窗函数应有最小的 B 和 A 及最大的 D。典型的窗函数及三项指标含义如图 3-4 所示。

在 MATLAB 中有生成矩形窗、三角窗、巴特列特窗、布莱克曼窗、切比雪夫窗、汉明窗、汉宁窗和凯撒窗的函数。下面画出其波形及幅频响应曲线并给出三项指标，同时给出 $\boldsymbol{w}$ 从 0 到 $N-1$ 和从 $-(N-1)/2$ 到 $(N-1)/2$ 的解析式，

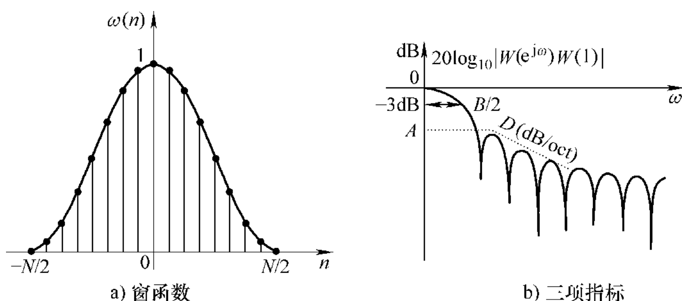


图 3-4 窗函数 **w** 及性能指标示意

它们分别以 $n=(N-1)/2$ 和 $n=0$ 为对称（只讨论 N 为奇数情形，给出的图形以 $n=0$ 对称）。

1. 矩形窗（Rectangle Window）

$$\boldsymbol{w}(n)=1,\ n=0,\ 1,\ \cdots,\ N-1$$

$$\boldsymbol{w}(n)=1,\ n=-(N-1)/2,\ \cdots,\ -1,\ 0,\ 1,\ \cdots\ (N-1)/2$$

其谱函数为

$$W(\mathrm{e}^{\mathrm{j}\omega})=\mathrm{e}^{-\mathrm{j}\frac{(N-1)\omega}{2}}\frac{\sin\left(\frac{N\omega}{2}\right)}{\sin\left(\frac{\omega}{2}\right)}$$

实现 **w** 及谱 **W** 的 MATLAB 代码如下：

```
functioncomWins(T)
% this function is to draw the figure of different windows as rectangle(r)
% and triangle(t)consine(c)hanning(h)hamming(m)blackman(b)papulis(p)
% both in time and frequency domain

ifnargin = =0
    N =17;T ='r';wres =2* pi/N;wsample =pi/N^4;
    pc =struct;
end
rn1 =(N -mod(N,2))/2;rn2 =1 -mod(N,2);
nl = -rn1 -rn2;nr =rn1 -rn2;

switch T
    case'r'
    pc.wL = -pi/2;pc.wR = -pc.wL;
    w =pc.wL:wsample:pc.wR;w(w = =0) =eps;
    tfig =ones(1,nr -nl +1);
    pc.tfmin =min(tfig);pc.tfmax =max(tfig);pc.norv =N;
```

```
    famp = sin(N* w/2)./sin(w/2);famp = famp/pc.norv;
    fampdb = 20* log10(abs(famp));
    pc.dbL = -40;pc.Bdev = 0.01;pc.Adev = 0.0001;
    pc.lobe1L = 2* pi/N - pc.wL;pc.lobe1R = 4* pi/N - pc.wL;
    otherwise
       disp('sorry! no such window.');
end
B = max(w(abs(fampdb + 3) < pc.Bdev))* 2/(2* pi/N)
lobe1LR = [pc.lobe1L,pc.lobe1R];w1LR = lobe1LR/wsample;
A = max(fampdb(w1LR(1):w1LR(2))),w1 = max(w(fampdb == A));
A1 = min(fampdb(abs(w - w1* 2) < pc.Adev));D = A1 - A

figure(1);
subplot(2,2,[1,2]),h = stem([nl:nr],tfig,'k--','MarkerSize',6);
set(h,'MarkerFaceColor','k');
set(gca,'XLim',[nl-2,nr+2],'YLim',[pc.tfmin* 0.9,pc.tfmax* 1.02]);
grid on;xlabel('n');ylabel('normalized amp');title('{\it \bfw}');
set(gca,'FontSize',16,'XMinorTick','on','YMinorTick','on');
subplot(2,2,3),plot(w,famp,'k','LineWidth',2);
grid on;xlabel('\omega(rad/s)');
ylabel('normalized amp');title('W');
set(gca,'FontSize',16,'XMinorTick','on','YMinorTick','on');
subplot(2,2,4),plot(w,fampdb,'k','LineWidth',2);
hold on,plot(w1,A,'k.',w1* 2,A1,'k.','MarkerSize',26);
line([w1,w1* 2],[A,A1],'Color','k','LineWidth',2);
set(gca,'YLim',[pc.dbL,0],'XMinorTick','on','YMinorTick','on');
grid on;xlabel('\omega(rad/s)');ylabel('attenuation(dB)');
title('spectrum property');
set(gca,'FontSize',16);
```

运行结果如图 3-5 所示。三项衡量指标分别为：$B=0.8872\Delta\omega$，$A=-13.1468\text{dB}$，$D=-13.1777\text{dB/oct}$。

2. 三角窗（Bartlett Window）

$$w(n)=\begin{cases}\dfrac{n}{(N-1)/2} & n=0,\ 1,\ \cdots,\ (N-1)/2\\ w(N-1-n) & n=(N-1)/2,\ \cdots,\ N-1\end{cases}$$

或 $w(n)=1-\dfrac{|n|}{(N-1)/2}\quad n=-(N-1)/2,\ \cdots,\ -1,\ 0,\ 1,\ \cdots,\ (N-1)/2$

其谱函数为

$$W(\mathrm{e}^{\mathrm{j}\omega})=\frac{2}{N-1}\mathrm{e}^{-\mathrm{j}(N-1)\omega/2}\left[\frac{\sin((N-1)\omega/4)}{\sin(\omega/2)}\right]^2$$

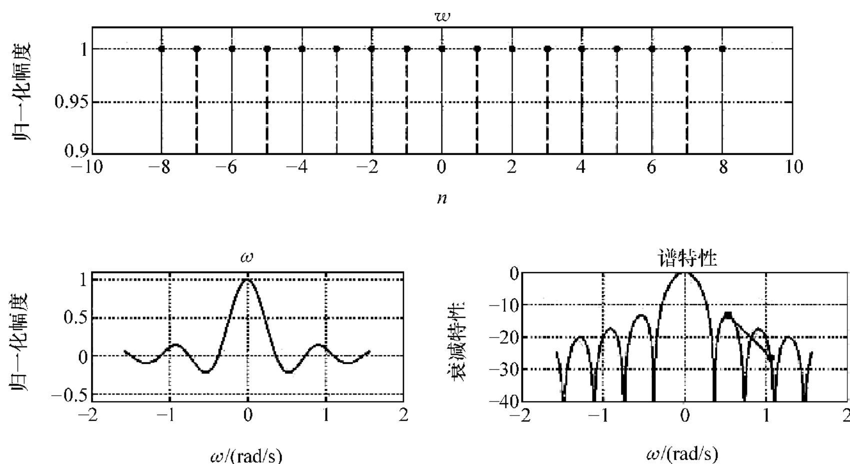


图 3-5　矩形窗及谱图

实现 $\boldsymbol{w}$ 及谱 $\boldsymbol{W}$ 的 MATLAB 代码如下（只列出不同部分）：

```
case 't'
pc.wL = -pi;pc.wR = -pc.wL;
w =pc.wL:wsample:pc.wR;w(w = =0) =eps;
tfigr = [(N -1)/2: -1:0]/((N -1)/2);
tfig = [fliplr(tfigr),tfigr(2:end)];
pc.tfmin =min(tfig);pc.tfmax =max(tfig);
pc.norv =N^2/4;
famp = (sin(N* w/4)./sin(w/2)).^2;
famp =famp/pc.norv;
fampdb =20* log10(abs(famp));
pc.dbL = -60;pc.Bdev =0.01;pc.Adev =0.001;
pc.lobe1L =4* pi/N -pc.wL;
pc.lobe1R =8* pi/N -pc.wL;
```

运行结果如图 3-6 所示。三项衡量指标分别为：$B=1.2837\Delta\omega$，$A=-25.7022\mathrm{dB}$，$D=-25.8864\mathrm{dB/oct}$。

3. 余弦窗（Cosine Window）

$$\boldsymbol{w}(n)=\sin[n\pi/(N-1)]\quad n=0,\ 1,\ 2,\ \cdots,\ N-1$$

或 $\boldsymbol{w}(n)=\cos[n\pi/(N-1)]\quad n=-(N-1)/2,\ \cdots,\ -1,\ 0,\ 1,\ \cdots,\ (N-1)/2$

其谱函数为

$$W(\mathrm{e}^{\mathrm{j}\omega})=\frac{1}{2}\mathrm{e}^{-\mathrm{j}\frac{(N-1)\omega}{2}}\left[U\left(\omega-\frac{2\pi}{N-1}\right)+U\left(\omega+\frac{2\pi}{N-1}\right)\right],\ U(\omega)=\mathrm{e}^{\mathrm{j}\frac{\omega}{2}}\frac{\sin(N\omega/2)}{\sin(\omega/2)}$$

实现 $\boldsymbol{w}$ 及谱 $\boldsymbol{W}$ 的 MATLAB 代码如下（只列出不同部分）：

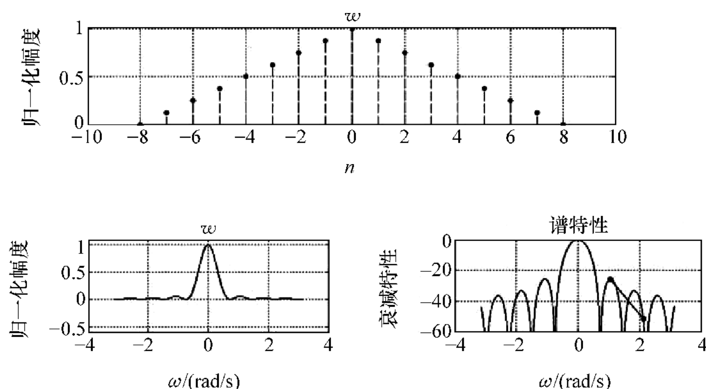


图 3-6　三角窗及谱图

```
case 'c'
pc.wL = -pi;pc.wR = -pc.wL;
w =pc.wL:wsample:pc.wR;w(w = =0) =eps;
n = -(N -1)/2:(N -1)/2;
tfig =cos(n* pi/(N -1));
pc.tfmin =min(tfig);pc.tfmax =max(tfig);
pc.norv =2/sin(pi/(N -1)/2);
wminus =w -pi/(N -1);
wplus =w +pi/(N -1);
Uminus =exp(j* wminus/2).* sin(N* wminus/2)./sin(wminus/2);
Uplus =exp(j* wplus/2).* sin(N* wplus/2)./sin(wplus/2);
famp =Uminus +Uplus
fpha =exp( -j* (N -1)/2* w);famp =famp.* fpha;
famp =abs(famp/pc.norv);
fampdb =20* log10(abs(famp));
pc.dbL = -45;pc.Bdev =0.01;pc.Adev =0.001;
pc.lobe1L =4* pi/N -pc.wL;
pc.lobe1R =8* pi/N -pc.wL;
```

运行结果如图 3-7 所示。三项衡量指标分别为：$B=1.2689\Delta\omega$，$A=-22.0788\text{dB}$，$D=-11.2495\text{dB/oct}$。

值得注意的是，$U[\omega-2\pi/(N-1)]$ 和 $U[\omega+2\pi/(N-1)]$ 的幅值没有共同零点，所以谱 W 值均大于零，只在两边无穷远处趋近于零。此外，$U(\omega)$ 含有相位信息，因此谱 W 整体不具备线性相位。其相位特性如图 3-8 所示。

4. 汉宁窗（Hanning Window）

$$\boldsymbol{w}(n)=\sin^2[n\pi/(N-1)]=0.5-0.5\cos[2n\pi/(N-1)]\quad n=0,\ 1,\ \cdots,\ N-1$$

或

$$\boldsymbol{w}(n)=\cos^2[n\pi/(N-1)]=0.5+0.5\cos[2n\pi/(N-1)]$$

$$n=-(N-1)/2,\ \cdots,\ -1,\ 0,\ 1,\ \cdots,\ (N-1)/2$$

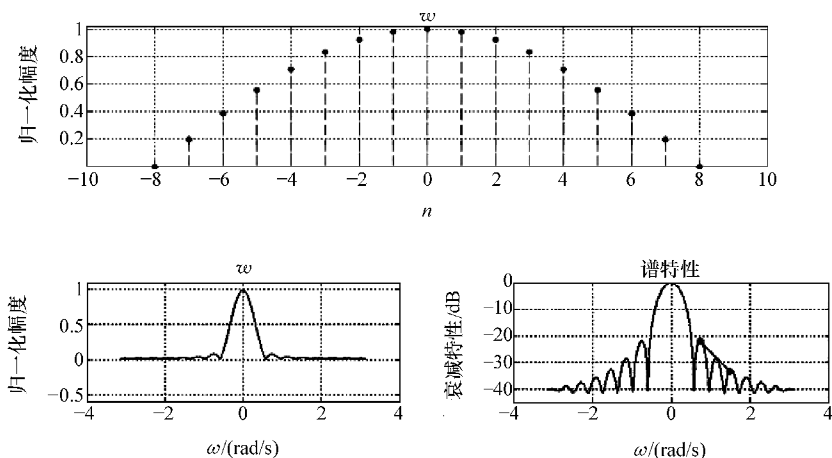


图 3-7　余弦窗及谱图

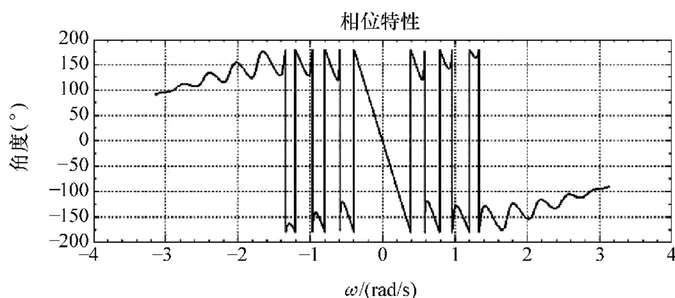


图 3-8　余弦窗谱相位特性

其谱函数为

$$W(e^{j\omega}) = 0.5U(\omega) + 0.25\{U[\omega - 2\pi/(N-1)] + U[\omega + 2\pi/(N-1)]\}$$

实现 ***w*** 及谱 ***W*** 的 MATLAB 代码如下（只列出不同部分）：

```
case 'h'
pc.wL = -pi;pc.wR = -pc.wL;
w = pc.wL:wsample:pc.wR;w(w = =0) = eps;
n = -(N-1)/2:(N-1)/2;
tfig = 0.5 +0.5* cos(2* n* pi/(N-1));
pc.tfmin = min(tfig);pc.tfmax = max(tfig);
pc.norv = 0.5* (N + sin(N* pi/(N-1))/sin(pi/(N-1)));
wminus = w -2* pi/(N-1);wplus = w +2* pi/(N-1);
U = exp(j* w/2).* sin(N* w/2)./sin(w/2);
Uminus = exp(j* wminus/2).* sin(N* wminus/2)./sin(wminus/2);
Uplus = exp(j* wplus/2).* sin(N* wplus/2)./sin(wplus/2);
```

```
famp=0.5*U+0.25*(Uminus+Uplus);
famp=abs(famp/pc.norv);
fampdb=20*log10(abs(famp));
pc.dbL=-70;pc.Bdev=0.01;pc.Adev=0.001;
pc.lobe1L=4*pi/N-pc.wL;pc.lobe1R=8*pi/N-pc.wL;
```

运行结果如图 3-9 所示。三项衡量指标分别为：$B=1.5260\Delta\omega$，$A=-31.4167\text{dB}$，$D=-17.7265\text{dB/oct}$。同余弦窗的情形相似，其谱没有出现零点。其相位特性如图 3-10 所示。

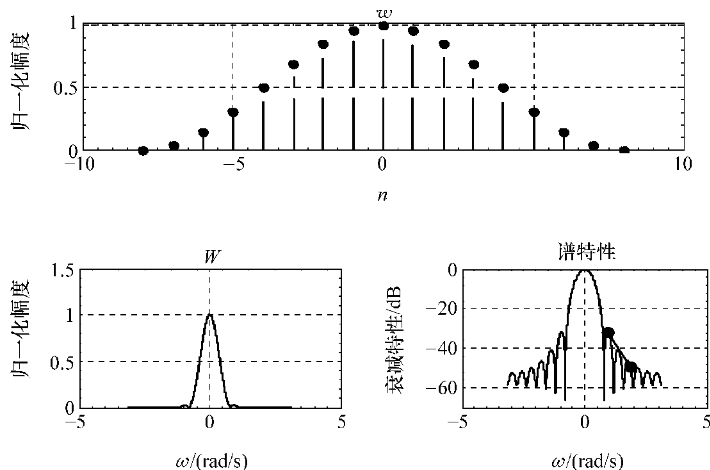


图 3-9　汉宁窗及谱图

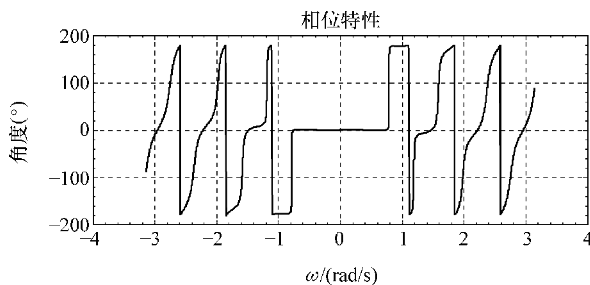


图 3-10　汉宁窗谱相位特性

5. 汉明窗（Hamming Window）

$$w(n)=0.54-0.46\cos\{2n\pi/(N-1)\}n=0,1,\cdots,N-1$$

或
$$w(n)=0.54+0.46\cos[2n\pi/(N-1)]$$
$$n=-(N-1)/2,\cdots,-1,0,1,\cdots,(N-1)/2$$

其谱函数为

$$W(e^{j\omega})=0.54U(\omega)+0.23[U[\omega-2\pi/(N-1)]+U[\omega+2\pi/(N-1)]]$$

实现 ***w*** 及谱 ***W*** 的 MATLAB 代码如下（只列出不同部分）：

```
case 'm'
pc.wL = -pi;pc.wR = -pc.wL;
w =pc.wL:wsample:pc.wR;w(w = =0) =eps;
n = -(N -1)/2:(N -1)/2;
tfig =0.54 +0.46* cos(2* n* pi/(N -1));
pc.tfmin =min(tfig);pc.tfmax =max(tfig);
pc.norv =0.54* N +0.46* (sin(N* pi/(N -1))/sin(pi/(N -1)));
wminus =w -2* pi/(N -1);wplus =w +2* pi/(N -1);
U =exp(j* w/2).* sin(N* w/2)./sin(w/2);
Uminus =exp(j* wminus/2).* sin(N* wminus/2)./sin(wminus/2);
Uplus =exp(j* wplus/2).* sin(N* wplus/2)./sin(wplus/2);
famp =0.54* U +0.23* (Uminus +Uplus);
famp =abs(famp/pc.norv);
fampdb =20* log10(abs(famp));
pc.dbL = -60;pc.Bdev =0.01;pc.Adev =0.001;
pc.lobe1L =5* pi/N -pc.wL;pc.lobe1R =6* pi/N -pc.wL;
```

运行结果如图 3-11 所示。三项衡量指标分别为：$B=1.3495\Delta\omega$，$A=-39.0547\text{dB}$，$D=-5.6532\text{dB/oct}$。同汉宁窗的情形相似，其谱没有出现零点。其相位特性如图 3-12所示。

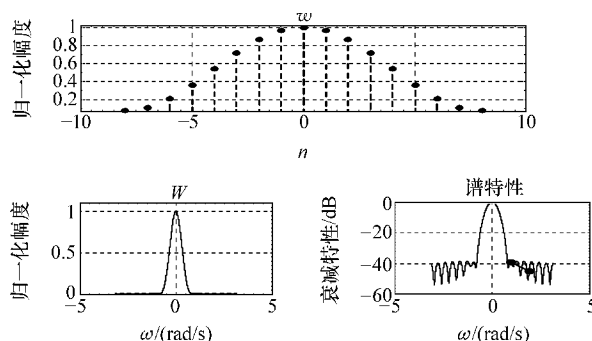


图 3-11　汉明窗及谱图

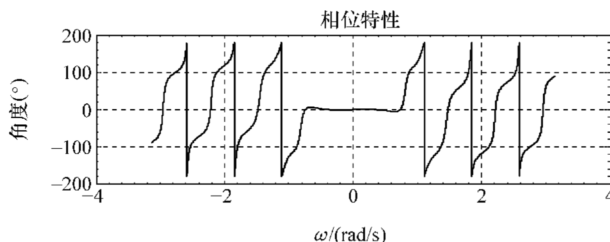


图 3-12　汉明窗谱相位特性

6. 布莱克曼窗（Blackman Window）

$w(n)=0.42-0.5\cos[2n\pi/(N-1)]+0.08\cos[4n\pi/(N-1)] \quad n=0, 1, \cdots, N-1$

或者　$w(n)=0.42+0.5\cos[2n\pi/(N-1)]+0.08\cos[4n\pi/(N-1)]$

$n=-(N-1)/2, \cdots, -1, 0, 1, \cdots, (N-1)/2$

其谱函数为

$$W(e^{j\omega})=0.42U(\omega)+0.25\left[U\left(\omega-\frac{2\pi}{N-1}\right)+U\left(\omega+\frac{2\pi}{N-1}\right)\right]+$$
$$0.04\left[U\left(\omega-\frac{4\pi}{N-1}\right)+U\left(\omega+\frac{4\pi}{N-1}\right)\right]$$

实现 **w** 及谱 **W** 的 MATLAB 代码如下（只列出不同部分）：

```
case 'b'
pc.wL = -pi;pc.wR = -pc.wL;
w = pc.wL:wsample:pc.wR;w(w == 0) = eps;
n = -(N-1)/2:(N-1)/2;
tfig = 0.42 + 0.5* cos(2* n* pi/(N-1)) + 0.08* cos(4* n* pi/(N-1));
pc.tfmin = min(tfig);pc.tfmax = max(tfig);
pc.norv = 0.42* N + 0.5* (sin(N* pi/(N-1))/sin(pi/(N-1))) +
0.08* (sin(2* N* pi/(N-1))/sin(2* pi/(N-1)));
wminus1 = w - 2* pi/(N-1);wplus1 = w + 2* pi/(N-1);
U = exp(j* w/2).* sin(N* w/2)./sin(w/2);
Uminus1 = exp(j* wminus1/2).* sin(N* wminus1/2)./sin(wminus1/2);
Uplus1 = exp(j* wplus1/2).* sin(N* wplus1/2)./sin(wplus1/2);
wminus2 = w - 4* pi/(N-1);wplus2 = w + 4* pi/(N-1);
Uminus2 = exp(j* wminus2/2).* sin(N* wminus2/2)./sin(wminus2/2);
Uplus2 = exp(j* wplus2/2).* sin(N* wplus2/2)./sin(wplus2/2);
famp = 0.42* U + 0.25* (Uminus1 + Uplus1) + 0.04* (Uminus2 + Uplus2);
famp = abs(famp/pc.norv);
fampdb = 20* log10(abs(famp));
pc.dbL = -80;pc.Bdev = 0.01;pc.Adev = 0.001;
pc.lobe1L = 6* pi/N - pc.wL;pc.lobe1R = 8* pi/N - pc.wL;
```

运行结果如图 3-13 所示。三项衡量指标分别为：$B=1.7439\Delta\omega$，$A=-55.2107\text{dB}$，$D=-8.2572\text{dB/oct}$。同汉明窗的情形相似，其谱没有出现零点。其相位特性如图 3-14 所示。

7. 帕普利斯窗（Papoulis Window）

$$w(n)=\frac{1}{\pi}\left|\sin\left(\frac{2n\pi}{N-1}\right)\right|-\left(1-\frac{2\,|n-(N-1)/2|}{N-1}\right)\cos\left(\frac{2n\pi}{N-1}\right) \quad n=0, 1, \cdots, N-1$$

或者
$$w(n)=\frac{1}{\pi}\left|\sin\left(\frac{2n\pi}{N-1}\right)\right|+\left(1-\frac{2\,|n|}{N-1}\right)\cos\left(\frac{2n\pi}{N-1}\right)$$

$$n = -\frac{N-1}{2}, \ \cdots, \ -1, \ 0, \ 1, \ \cdots, \ \frac{N-1}{2}$$

其谱函数为

$$W(e^{j\omega}) = 32(N-1)\pi \frac{1+\cos\left[\frac{(N-1)\omega}{2}\right]}{[(N-1)^2\omega^2 - 4\pi^2]^2}$$

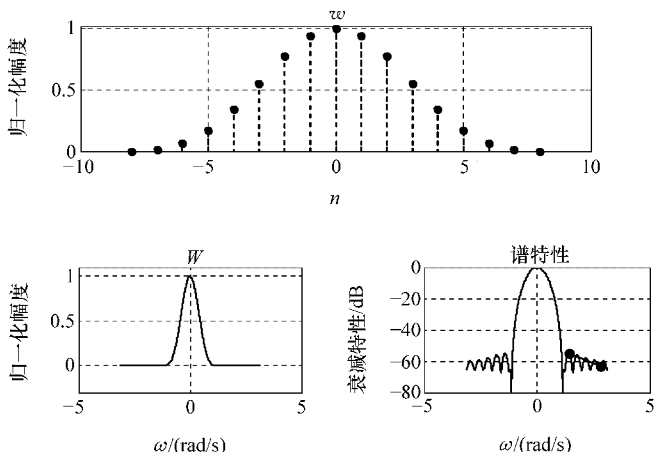


图 3-13 布莱克曼窗及谱图

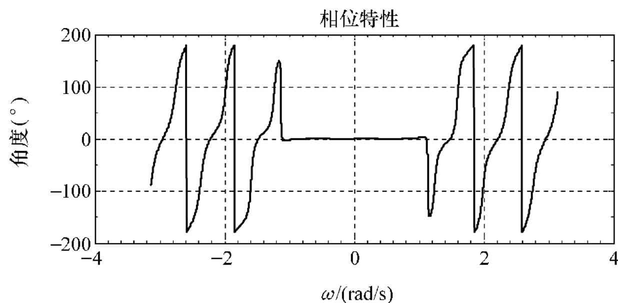


图 3-14 布莱克曼窗谱相频特性

实现 ***w*** 及谱 ***W*** 的 MATLAB 代码如下（只列出不同部分）：

```
case 'p'
pc.wL = -pi;pc.wR = -pc.wL;
w =pc.wL:wsample:pc.wR;w(w = =0) =eps;
n = -(N -1)/2:(N -1)/2;
tfig =abs(sin(2* n* pi/(N -1)))/pi +...
   (1 -2* abs(n)/(N -1)).* cos(2* n* pi/(N -1));
pc.tfmin =min(tfig);
```

```
pc.tfmax =max(tfig);
pc.norv =4* N/pi^3;
mp =(1 +cos((N -1)* w/2))./((N -1)^2* w.^2 -4* pi^2).^2;
famp =32* (N -1)* pi* mp;
famp =famp/pc.norv;
fampdb =20* log10(abs(famp));
pc.dbL = -80;pc.Bdev =0.01;pc.Adev =0.001;
pc.lobe1L =6* pi/N -pc.wL;
pc.lobe1R =8* pi/N -pc.wL;
```

运行结果如图 3-15 所示。三项衡量指标分别为：$B=1.6453\Delta\omega$，$A=-46.5291\text{dB}$，$D=-29.1660\text{dB/oct}$。帕布里斯窗谱出现零点且具有恒定零相位特性。

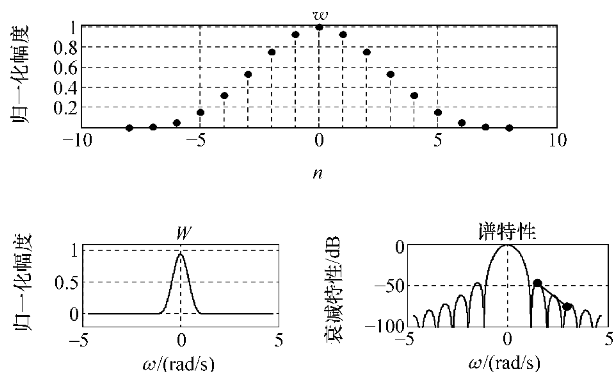


图 3-15　帕布里斯窗及谱图

综合以上，矩形窗谱具有最窄的主瓣，但旁瓣峰值也最大；三角窗谱和余弦窗谱主瓣宽度稍宽，但具有较小旁瓣和较大衰减速度。整理各种窗谱指标见表 3-1。

表 3-1　7 种窗的 4 项指标

	主 瓣 宽 度	$B(\Delta\omega)$	A/dB	D/(dB/oct)
矩形窗	$4\pi/N$	0.89	-13.15	-13.18
三角窗	$8\pi/N$	1.28	-25.70	-25.89
余弦窗	$6\pi/N$	1.27	-22.08	-11.25
汉宁窗	$8\pi/N$	1.53	-31.42	-17.73
汉明窗	$8\pi/N$	1.35	-39.05	-5.65
布莱克曼窗	$12\pi/N$	1.74	-55.21	-8.26
帕普利斯窗	$12\pi/N$	1.65	-46.53	-29.17

A 与 B 或者 D 与 B 在限定算法中基本不能同时改善，为综合衡量窗谱三项指标而定义两个变量：每频率分辨率的最小旁瓣衰减 A/B（单位是 dB/rad）和每频率分辨率的衰减速度（单位是 dB/oct/rad）。它们也分别称为窗口速度和窗口加速度，即

$$\begin{aligned} s_w &= A/B \\ a_w &= D/B \end{aligned} \tag{3-5}$$

表 3-1 中七种窗的 a_w 和 s_w 对比图如图 3-16 所示。

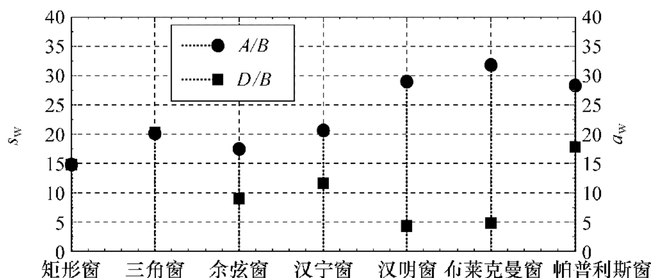


图 3-16　七种窗 a_w 和 s_w

实现图 3-16 的 MATLAB 代码如下：

```
function AD2B
t ={'rectangle';'triangle';'cosine';'hanning';...
'hamming';'blackman';'populis'};
B =[0.89 1.28 1.27 1.53 1.35 1.74 1.65];% less is better
A =[13.15 25.7 22.08 31.42 39.05 55.21 46.53];% more is better
D =[13.18 25.89 11.25 17.73 5.65 8.26 29.17];% more is better
AB =A./B;% attenuation rate;dB/rad
DB =D./B;% attenuationaccelation;dB/oct/rad
[ax,hL,hR] =plotyy([1:length(t)],A./B,[1:7],D./B,'stem','stem');
set(hR,'Marker','s');set([hL,hR],'MarkerSize',10);
set([hL,hR],'Color','k','LineWidth',2,'MarkerFaceColor',...
'k','LineStyle',':');
set([ax(1),ax(2)],'FontSize',16,'XMinorTick','on','YMinorTick','on');
set([ax(1),ax(2)],'XTickLabel',t,'YTick',[0:5:40]);
set(get(ax(1),'YLabel'),'String','\its_w','FontSize',16);
set(get(ax(2),'YLabel'),'String','\ita_w','FontSize',16);
legend('A/B','D/B');grid on;
```

观察图 3-16 可以发现，布莱克曼窗谱具有最高窗口速度但加速度最小，矩形窗谱窗口速度最小，但加速度大于余弦窗和汉宁窗，三角窗和帕普利斯窗谱的窗口速度和加速度均良好。如果把计算复杂度考虑进去，三角窗是 FIR 滤波器窗

函数设计法中比较好的选择。除上面常见的 7 种窗外，还有 Kaiser 窗、Chebyshev 窗、Cauchy 窗、Gaussian 窗等，它们都可对滤波器性能起到一定的改善作用。

为了得到较好性能的滤波器，在设计时选择合适的窗函数是必要和非常重要的问题。但由于滤波器过渡带带宽与阻带最小衰减相互矛盾，在一定程度上，任何窗函数只能牺牲其中一项指标来换取另一项的改善，同时，滤波器的设计应该结合信号特性和处理目的来进行。

3.1.3　apDW 频率取样误差分析

由第 2 章中内容可知，apNW 和 apSW 的频偏向量 $\boldsymbol{E}_n$元素恒为 N，其传输特性在频率采样点与 $\boldsymbol{H}$ 对应取样相同。但 apDW 的 $\boldsymbol{E}_n$值并不恒定，其浮动程度反映了传输特性在各频率抽样点与 $\boldsymbol{H}$ 的偏差。以上节介绍的六种窗（矩形窗除外）分别作为 apDW 的基窗时（归一化）$\boldsymbol{E}_n$曲线如图 3-17 所示。

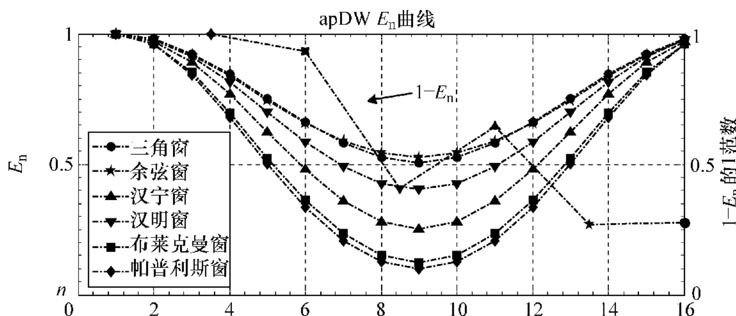


图 3-17　apDW 六种基窗的 $\boldsymbol{E}_n$

由图 3-17 可看出，六种基窗频偏均具有“盆”形状，余弦窗频偏稍低于三角窗，两者频偏最小。实现上述结果的 MATLAB 代码如下：

```
function drawEn(N)
% this function is to draw En and its distance to 1
if nargin = =0
   N =16;
end
funs = {'triang(N)','fcosw(N)','hann(N)','hamming(N)','blackman(N)','bohman-
win(N)'};
n = length(funs);ws = zeros(2* N -1,n);En = zeros(N,n);
for i =1:n
   ws(:,i) = conv(eval(funs{i}),eval(funs{i}));
end
En = [ws(N,:);ws(1:N -1,:) +ws(N +1:2* N -1,:)];
```

```
divEn = repmat(En(1,:),N,1);En = En./divEn;onesEn = ones(size(En));
norm1En = sum((onesEn - En).^2);norm1En = norm1En/max(norm1En);
markers = ['o';'p';'^';'v';'s';'d'];
[ax,h1,h2] = plotyy([1:N],En,[N+2.5-2.5*n:2.5:N],fliplr(norm1En));
set([h1;h2],'Color','k','LineWidth',2,'LineStyle','-.');
set(ax,'FontSize',16,'XMinorTick','on','YMinorTick','on');
grid on;hold on;xlabel('n');ylabel(ax(1),'En');
ylabel(ax(2),'1-norm of 1-En');title('apDW En Curve');
for i =1:n
  set(h1(i),'Marker',markers(i),'MarkerSize',6,'MarkerFaceColor','k');
plot(N-2.5*i+2.5,norm1En(i),'k','Marker',markers(i),'MarkerSize',8,
'MarkerFaceColor','k');
end
legend('triang','cos','hann','hamm','blackman','populis');
function y = fcosw(N)
n =0:N-1;
y = (sin(n*pi/(N-1)))';
```

余弦窗的 $\boldsymbol{E}_n$ 曲线凹陷幅度随 N 不同而稍有变化，且 N 相当大时“盆底”幅值收敛于0.64。余弦窗阶数 N 从 8 增大至 64 时的 $\boldsymbol{E}_n$ 曲线变化趋势如图 3-18 所示。

三角窗、汉宁窗、汉明窗、布莱克曼和帕普利斯窗的频偏向量 $\boldsymbol{E}_n$ 随 N 化规律与余弦窗类似，在阶数相当大时分别收敛于 0.5、0.33、0.47、0.18 和 0.15，再次证明了余弦窗最小 $\boldsymbol{E}_n$ 的恒定性。实现上述结果的 MATLAB 代码如下：

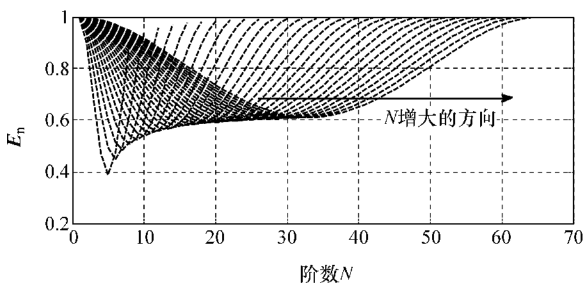


图 3-18　余弦 apDW 系统 $\boldsymbol{E}_n$ 随阶数变化曲线

```
function En2N(N1,N2)
% this function is to draw En curve changing with order N
ifnargin = =0
   N1 =4096*2;N2 =N1 +8;
end
funs = {'triang(N)','fcosw(N)','hann(N)','hamming(N)',...
   'blackman(N)','bohmanwin(N)'};
m =0;n = length(funs);
for N =N1:2:N2
   ws = zeros(2*N-1,n);m =m +1;
   for i =1:n
```

```
        ws(:,i) = conv(eval(funs{i}),eval(funs{i}));
    end
    En = [ws(N,:);ws(1:N -1,:) + ws(N +1:2* N -1,:)];
divEn = repmat(En(1,:),N,1);En = En./divEn;
    plot(En(:,6),'k - -','LineWidth',2),hold on;
end
set(get(gca,'XLabel'),'String','order N');
set(get(gca,'YLabel'),'String','En');grid on;

function y = fcosw(N)
n = 0:N -1;
y = (sin(n* pi/(N -1)))';
```

以上说明了不同基窗形成的卷积窗 ***C*** 作用于系统 ***H***，对应的幅频特性的频偏误差也不同。例如 $N=32$ 且截止频率为 $\pi/2$ 的 apDW 低通滤波器，分别以上述 6 种窗为基窗，分析并比较在采样点处的误差大小。将半带滤波器分成通带、过渡带和阻带 3 部分进行观察，曲线如图 3-19 所示。

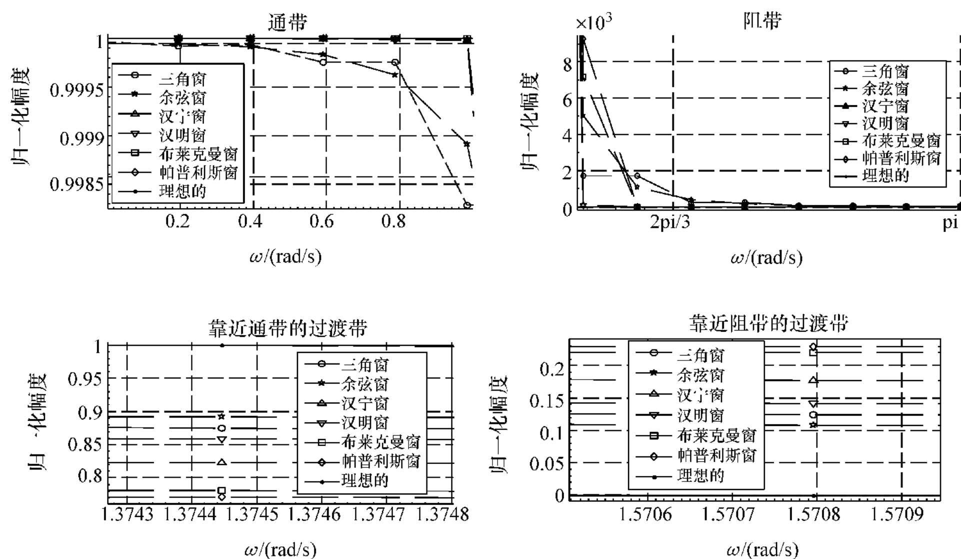


图 3-19　六基窗 apDW 频率采样误差分布

六基窗 apDW 频率采样与理想频率采样之间的误差见表 3-2。

表 3-2　频率采样与理想频率采样之间的误差

三　角　窗	余　弦　窗	汉　宁　窗	汉　明　窗	布莱克曼窗	帕普利斯窗
0. 2493	0. 2164	0. 3542	0. 2827	0. 4392	0. 4579

由表3-2可以看出，余弦窗对应的特性与 ***H*** 距离最小，三角窗稍大，帕普利斯窗窗距离最大。余弦窗这种优势基本上都是在过渡点上获得，而在通带和阻带内频率采样误差都大于三角窗。实现上述功能的 MATLAB 代码如下：

```
functionfreErrs(N)
% this function is to analyze the frequency sampling errors from apNW with
% 6 kinds of base window
ifnargin = =0
    N =32;fstop =pi/2;
end
H =zeros(N,1);M =round((N -2 +mod(N,2))/2* fstop/pi);
H(1:1 +M) =1;H(N -M +1:N) =1;h =ifft(H,N);
funs ={'triang(N)','fcosw(N)','hann(N)','hamming(N)',...
    'blackman(N)','bohmanwin(N)'};
n =length(funs);
for i =1:n
    C(:,i) =conv(eval(funs{i}),eval(funs{i}));
end
En =[C(N,:);C(1:N -1,:) +C(N +1:2* N -1,:)];
normfactor =repmat(En(1,:),N,1);En =En./normfactor;
hexpand =repmat(h,1,n);
hC =hexpand.* En;% dimension =N* n
w =0:2* pi/N:2* pi -2* pi/N;nw =[0:N -1]'* w;
expnw =exp( -j* nw);wlen =length(w);
for i =1:n
    hCexpand =repmat(hC(:,i),1,wlen);
    HW(i,:) =abs(sum(hCexpand.* expnw));
end
markers =['o';'p';'^';'v';'s';'d'];
for m =1:n
    hold on,h =plot(w,HW(m,:),'k - -','LineWidth',0.5);
    set(h,'Marker',markers(m),'MarkerSize',6);
    freErrs(m) =sqrt(sum((HW(m,:) -H').^2))
end
plot(w,H,'k - -.','MarkerSize',6);
set(gca,'XMinorTick','on','YMinorTick','on','FontSize',16);
xtl ={'0','pi/3','2pi/3','pi','4pi/3','5pi/3','2pi'};
set(gca,'XTick',[0:pi/3:2* pi],'XTickLabel',xtl);
% set(gca,'XLim',[2* pi/N* (M)2* pi/N* (M +3)]);
xlabel('\omega(rad/s)');ylabel('normilized amp');grid on;
legend('triangle','consine','hanning','hamming','blackman','populis','idea');
```

```
function y = fcosw(N)
n = 0:N - 1;
y = (sin(n* pi/(N - 1)))';
```

此外，频率采样误差也随阶数 N 的增大而增大。由生成图 3-18 结果的实验可知，当 N 相当大时的频率采样误差也将趋近于稳定。截止频率为 $\pi/2$ 的 apDW，频率采样误差随 N 变化的曲线如图 3-20 所示。

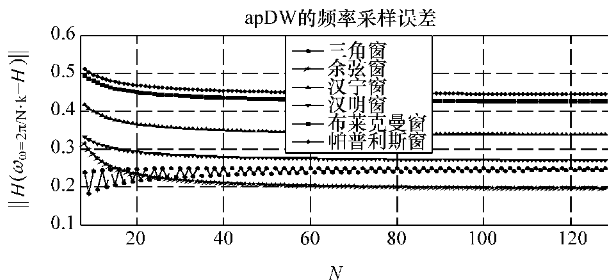


图 3-20　六基窗 apDW 频率采样误差随阶数变化曲线

基窗为三角窗时的频率采样误差出现“奇偶振荡”现象，且随 N 增大而增大并趋于平稳收敛于 0. 2501。其他基窗对应的频率采样误差均满足递减规律，其中余弦窗具有最低误差和最大的下降速率且收敛于 0. 1912，其次是汉明窗收敛于 0. 2672，第三是汉宁窗收敛于 0. 3346，第四是布莱克曼窗收敛于 0. 4221，帕普利斯窗对应的频率采样误差最大，收敛于 0. 4413。由图 3-20 还可以看出，当阶数大于 80 后的误差水平趋于稳定，从而提供了选定阶数的一种限定基准。而且由图 3-19 可知频率采样距离分布不均匀，所以并不能通过任意增大 N 以达到理想特性。另外，频率采样距离的大小也会受通带或阻带宽度的影响（实现代码基本与 freErrs 相同，不再列出）。

3. 2　基于 LMS 准则的基窗设计算法

3. 1 节在对 6 种基窗函数分析的基础上，指出以余弦窗为基窗的 apDW 具有最小的频率采样误差。这种比较是有局限性的，首先比较是在固定基窗模式下进行，对于其他未知形状的窗并没有进行推论或衡量。此外，在对比时，对于频带中其他频率成分的误差并不能同时保证为最小或提出充分条件。为寻找更为理想的基窗函数，借助 LMS 算法思想提出了“基于 LMS 准则以 apSW 为模型的基窗函数设计方法”。即将 $[0, \pi]$ 间的频率采样点 ω_i 的取值定义为随机过程，通过最小频采误差为收敛准则构造收敛模型，设定 apSW 具有线性相位且 N 为偶数（N 为奇数时可作类似分析）。

式（2-51）表示的幅频特性可表示成如下表达式：

$$H_g(\omega) = \sum_{n=0}^{N-1} \boldsymbol{c}_n \cos(n\omega) = \langle \boldsymbol{C}, \boldsymbol{S} \rangle \tag{3-6}$$

式中，$S(n\omega) = \begin{cases} h(0) & n=0 \\ 2h(n)\cos(\omega n) & n=1, 2, \cdots, N-1 \end{cases}$。把式（2-57）结论表达式 c_n 代入上式并由基窗 $\boldsymbol{f}$ 的中心对称性得：

$$\begin{aligned} H_g(\omega) &= \sum_{k=0}^{N-1}\left[\sum_{n=k}^{N-1} f(n)S(k\omega)\right] \\ &= \sum_{n=0}^{N-1}\left[f(n)\sum_{k=0}^{n} S(k\omega)\right] \\ &= \sum_{n=0}^{N/2-1}\left[f(n)\sum_{k=0}^{n} S(k\omega)\right] + \sum_{n=0}^{N/2-1}\left[f(N-1-n)\sum_{k=0}^{N-1-n} S(k\omega)\right] \\ &= \sum_{n=0}^{N/2-1}\left\{f(n)\left[\sum_{k=0}^{n} S(k\omega) + \sum_{k=0}^{N-1-n} S(k\omega)\right]\right\} \end{aligned} \tag{3-7}$$

式（3-7）表明，对于 N 阶线性相位 apSW，只需 $\boldsymbol{f}$ 半边信息就可得到幅频特性。重新定义基窗向量 $\boldsymbol{F}$ 如下：

$$\boldsymbol{F} = [f(0) \quad f(1) \quad \cdots \quad f(N/2-1)]^{\mathrm{T}} \tag{3-8}$$

并定义余弦向量为

$$\boldsymbol{S}_{\mathrm{m}} = [\cos_0(m) \quad \cos_1(m) \quad \cdots \quad \cos_{N-1}(m)]^{\mathrm{T}} \tag{3-9}$$

式中，$\cos_n(m) = \cos(n\omega_m)$　$n=0, 1, \cdots, N-1$；$m=0, 1, \cdots, M$，M 是实验样本数。构造冲击响应加权矩阵 $\boldsymbol{G}$ 如下：

$$\boldsymbol{G} = \begin{bmatrix} 2h(0) & 2h(1) & 2h(2) & 2h(3) & \cdots & 2h(N-3) & 2h(N-2) & 2h(N-1) \\ 2h(0) & 4h(1) & 2h(2) & 2h(3) & \cdots & 2h(N-3) & 2h(N-2) & 0 \\ 2h(0) & 4h(1) & 4h(2) & 2h(3) & \cdots & 2h(N-3) & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 2h(0) & 4h(1) & 4h(2) & 4h(3) & \cdots & 4h\left(\frac{N}{2}-1\right) & & \end{bmatrix}_{N/2\times N} \tag{3-10}$$

幅频响应矢量 $\boldsymbol{H}_g$ 为

$$\boldsymbol{H}_g = [H_g(\omega_0) \quad H_g(\omega_1) \quad \cdots \quad H_g(\omega_{M-1})]^{\mathrm{T}} \tag{3-11}$$

结合式（3-8）~式（3-11）重写式（3-7）如下：

$$\boldsymbol{H}_g(\omega_m) = \boldsymbol{F}^{\mathrm{T}} \cdot \boldsymbol{G} \cdot \boldsymbol{S} \quad m=0, 1, \cdots, M-1 \tag{3-12}$$

设 apSW 理想幅频响应矢量为 $\boldsymbol{H}_{\mathrm{i}}$。当样本点数为 M 时，$\boldsymbol{H}_{\mathrm{i}}$ 可表示如下：

$$\boldsymbol{H}_{\mathrm{i}}=[H_{\mathrm{i}}(\omega_0)\quad H_{\mathrm{i}}(\omega_1)\quad \cdots\quad H_{\mathrm{i}}(\omega_{M-1})]^{\mathrm{T}} \tag{3-13}$$

则 apSW 的实际幅频响应 $\boldsymbol{H}_{\mathrm{g}}$ 在 ω_m 处的误差 ε_m 为

$$\varepsilon_m=H_{\mathrm{i}}(\omega_m)-H_{\mathrm{g}}(\omega_m)=H_{\mathrm{i}}(\omega_m)-\boldsymbol{F}^{\mathrm{T}}\cdot\boldsymbol{G}\cdot\boldsymbol{S} \tag{3-14}$$

在区间 [0，π] 内选取 ω_m 方式任意，故可将 ω_m 看作是区间 [0，π] 内的随机变量，则它的函数 $\boldsymbol{H}_{\mathrm{i}}$ 和 $\boldsymbol{G}$ 满足平稳随机分布。误差 ε_m 也具有随机性，故可用均方误差 $\xi_m=E\{\varepsilon_m{}^2\}$ 作为判定准则，使其达到最小值对应的基窗向量 $\boldsymbol{F}$ 即在此准则下的优化基窗函数，$\boldsymbol{F}$ 可采用最陡下降法进行求解。根据式（3-14）可得到均方误差为

$$\begin{aligned}\xi_m&=E\{[H_{\mathrm{i}}(\omega_m)-\boldsymbol{F}^{\mathrm{T}}\boldsymbol{G}S)]^2\}\\&=E\{H_{\mathrm{i}}^2(\omega_m)-2\boldsymbol{F}^{\mathrm{T}}\boldsymbol{GS}H_{\mathrm{i}}(\omega_m)+\boldsymbol{F}^{\mathrm{T}}\boldsymbol{GSS}^{\mathrm{T}}\boldsymbol{G}^{\mathrm{T}}\boldsymbol{F}\}\\&=E\{H_{\mathrm{i}}^2(\omega_m)\}-2\boldsymbol{F}^{\mathrm{T}}\boldsymbol{G}E\{\boldsymbol{S}H_{\mathrm{i}}(\omega_m)\}+\boldsymbol{F}^{\mathrm{T}}\boldsymbol{G}\mathrm{E}\{\boldsymbol{S}\boldsymbol{S}^{\mathrm{T}}\}\boldsymbol{G}^{\mathrm{T}}\boldsymbol{F}\\&=E\{H_{\mathrm{i}}^2(\omega_m)\}-2\boldsymbol{F}^{\mathrm{T}}\boldsymbol{G}\boldsymbol{R}_{\mathrm{SH_i}}+\boldsymbol{F}^{\mathrm{T}}\boldsymbol{G}\boldsymbol{R}_S\boldsymbol{G}^{\mathrm{T}}\boldsymbol{F}\end{aligned} \tag{3-15}$$

式（3-15）中 $\boldsymbol{R}_{\mathrm{SH_i}}$ 和 $\boldsymbol{R}_{\mathrm{S}}$ 分别是 $\boldsymbol{S}_{\mathrm{m}}$ 与 $\boldsymbol{H}_{\mathrm{i}}$ 互相关和 $\boldsymbol{S}_{\mathrm{m}}$ 自相关矩阵，即

$$\boldsymbol{R}_{\mathrm{SH_i}}=E\{S_{\mathrm{m}}H_{\mathrm{i}}(\omega_m)\}=E\begin{pmatrix}\cos_0(m)H_{\mathrm{i}}(\omega_m)\\\cos_1(m)H_{\mathrm{i}}(\omega_m)\\\vdots\\\cos_{N-1}(m)H_{\mathrm{i}}(\omega_m)\end{pmatrix} \tag{3-16}$$

$$\boldsymbol{R}_{\mathrm{S}}=E\{\boldsymbol{S}_{\mathrm{m}}\boldsymbol{S}_{\mathrm{m}}^{\mathrm{T}}\}=E\begin{pmatrix}\cos_0(m)\cos_0(m)&\cos_0(m)\cos_1(m)&\cdots&\cos_0(m)\cos_{N-1}(m)\\\cos_1(m)\cos_0(m)&\cos_1(m)\cos_1(m)&\cdots&\cos_1(m)\cos_{N-1}(m)\\\vdots&\vdots&\vdots&\vdots\\\cos_{N-1}(m)\cos_0(m)&\cos_{N-1}(m)\cos_1(m)&\cdots&\cos_{N-1}(m)\cos_{N-1}(m)\end{pmatrix} \tag{3-17}$$

对式（3-15）运用最陡下降法得到递推公式为

$$\boldsymbol{F}_{m+1}=\boldsymbol{F}_m+\mu\varepsilon_m\boldsymbol{G}\cdot\boldsymbol{S}_{\mathrm{m}} \tag{3-18}$$

式中，$\boldsymbol{F}_m$ 表示第 m 次迭代得到的窗函数。为保证均方误差收敛，步长 μ 应满足 $0<\mu<\lambda_{\max}$，$\lambda_{\max}$ 是 $\boldsymbol{G}\boldsymbol{R}_{\mathrm{S}}\boldsymbol{G}^{\mathrm{T}}$ 的最大特征值，可证明其满足：

$$\lambda_{\max}<N\max_{0\leqslant i\leqslant N-1}\{h_i^2\} \tag{3-19}$$

式中，h_i 是向量 $\boldsymbol{h}$ 的第 i 个元素。N 阶 apSW 实验样本数 M 可由下式确定：

$$\frac{\pi}{N/2}=k\frac{\pi}{M-1}\Rightarrow M=kN/2+1 \tag{3-20}$$

式（3-20）中 k 是大于 1 的整数。这样保证在每两个频采之间有 $k-1$ 个实验样本，若控制这些点使其幅值逼近理想值，就可使均方误差收敛到误差限 $\varepsilon_{\max}$ 以下。对于阶跃点之间的样本插值，在不增加过渡点的前提下，将其归为阻带或通带范围均可。由 M 确定的实验频率为

$$\omega_m = \frac{\pi}{M-1}m \quad m=0,\ 1,\ 2,\ \cdots,\ M-1 \tag{3-21}$$

这样，以式（3-19）定义步长并按照式（3-18）进行迭代，直到基窗 $\boldsymbol{F}$ 使 $\xi_m \leqslant \varepsilon_{\max}$成立即可。以上过程可用图 3-21 构造。相应的算法流程如图 3-22 所示。

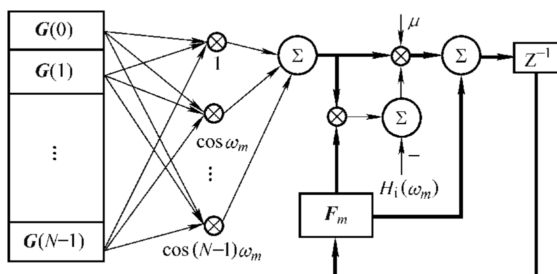


图 3-21　余弦基神经自适应网络

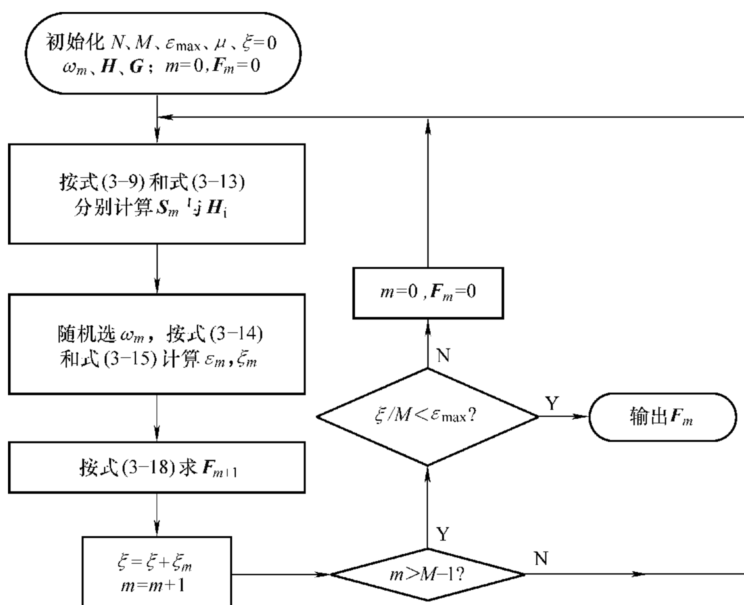


图 3-22　apSW 基窗 LMS 设计算法

按照流程图，设计截止频率为 π/4 的 32 阶低通 apSW 的窗函数 $\boldsymbol{F}$。取 $\boldsymbol{H}$ = [1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1]，半带中频采点数为（32 − 2）/2 + 2 = 17。在式（3-20）中令 $k=2$，样本点均匀地在频采点之间抽取，得到样本容量 $M=33$，则实验频率 $\omega_m = m\pi/32$（$m=0$，1，…，31）并令 $\varepsilon_{\max}=0.001$。以随机方式选取样本进行迭代，步长 $\mu=0.3951$，修正后等于 0.2371，初始基窗 $\boldsymbol{F}_0=0$。具体的 MATLAB 代码如下（省略画图部分）：

```
function[F] =lmsBW(N,wp,k,error)
%  this function is to find base window forapSW via LMS algorithm
ifnargin< =2
    N =32;wp =pi/4;k =2;error =0.001;
    F =zeros(1,N/2);m =1;times =1;sumerr =0;
end
clc,
M =k* N/2 +1;N2 =(N -mod(N,2))/2;
sones =round(N2* (wp/pi));H =zeros(1,N);
% below is to make out the step and impulse response weighted matrix Hc...
H(1:1 +sones) =1;H(N -sones +1:N) =1;h =ifft(H,N);
G =2* ones(N/2,N);
for i =2:N/2
    for j =2:i
      G(i,j) =4;
    end
    G(i,N +2 -i:N) =0;
end
G =G.* repmat(h,N/2,1);ustep =1/max(h.^2)/N;
% below is to compute the inflated ideal Hi by M sampling dots...
nd =k* (sones -1);Hi =zeros(1,M);Hi(1:nd) =1;
wm =[0:M -1].* pi/(M -1);col =[0:N -1]';
Sm =[0:N -1]'* wm;S =real(cos(Sm));
tic
while m <M +1
    j =mod(floor(rand* 10000),M) +1;GS =G* S(:,j);
    if sum(F) ~ =0
      F =F/sum(F)/2;
    end
    merr =Hi(1,j) -F* GS;sumerr =sumerr +abs(merr* merr);
    F =F +0.6* ustep* merr* GS';m =m +1;
    if m >M
      meanerr =sqrt(sumerr/M);
      ifmeanerr< =error
        break;
      else
        m =1;sumerr =0;times =times +1;
      end
    end
end
toc
```

```
times,meanerr,ustep
FW=[F,fliplr(F)];sWw=FW/sum(FW);
Cww=conv(sWw,ones(1,N));Cw=Cww(N:2*N-1);
w=0:pi/(M-1)^2:pi;tcos=w'*[1:N-1];COS=cos(tcos);
Hideal=2*sum(COS*diag(Cw(2:N).*h(2:N)),2)+Cw(1)*h(1);
```

在 CPU 主频为 2.6GHz 的计算机上运行共迭代 26462 次，耗时 22.267324s，得到 apSW 与理想幅频响应的均方误差为 6.5692×10^{-4}。收敛得到的基窗（归一化后）$\boldsymbol{F}$ = [0.0050 0.0046 0.0189 0.0229 0.0205 0.0255 0.0325 0.0327 0.0371 0.0401 0.0402 0.0409 0.0447 0.0448 0.0441 0.0455]。滤波器幅频响应、衰减特性和基窗 $\boldsymbol{F}$ 及对应的卷积窗 $\boldsymbol{C}$ 如图 3-23 所示。

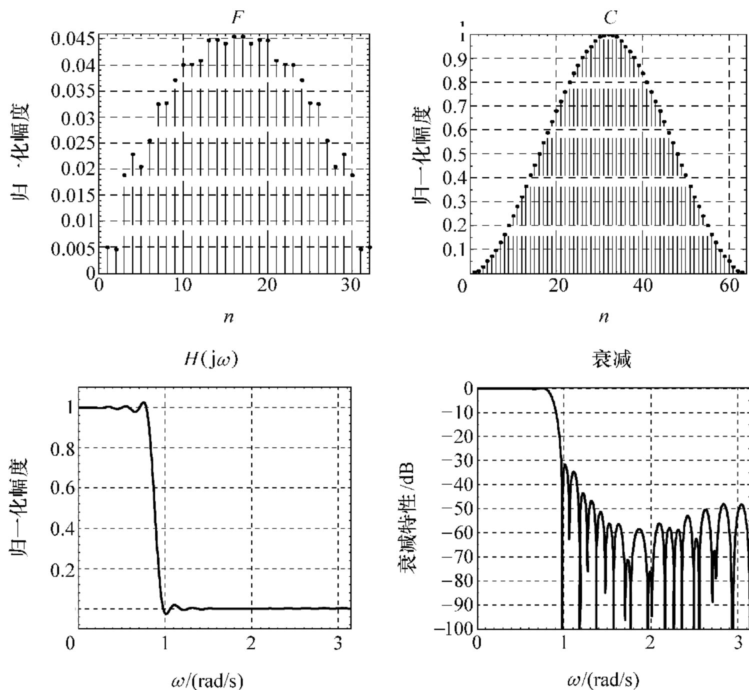


图 3-23 实验得到的基窗及 apSW 特性

由图 3-23 可看出，第一旁瓣最小衰减降至 −32.5dB，过渡带宽度约为 4.55π/63。将此参数对比表 3-1 各窗特性可知，通过此自适应算法得到的基窗在 B 和 A 两方面同时得到改善，窗口加速度 s_w 等于 143，大于现有的 7 种基窗。通带和阻带内的振荡幅度都得到较大程度抑制，但在过渡带附近出现了尖缝，这是由于实验中对处于阶跃点之间的一个样本只简单地将其归为通带而导致。这可以通过插入过渡点来得到改善，但过渡带也将加宽。基于基窗 $\boldsymbol{F}$ 的 apDW 特性如图 3-24 所示。

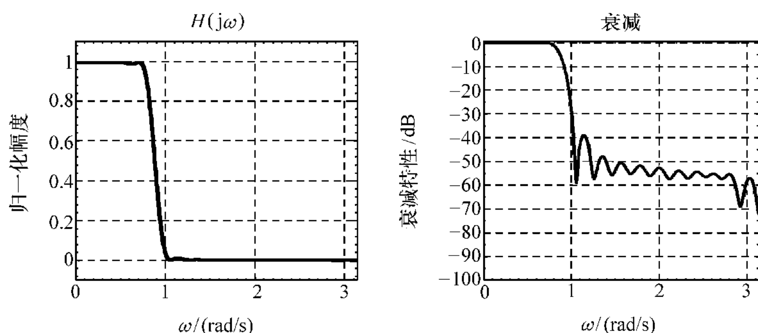


图 3-24　基于 **F** 的 apDW 特性

比较图 3-23 和图 3-24 可以看出，apDW 进一步降低了旁瓣最小衰减至 −38.9dB，过渡带宽度约为 7π/63，窗口加速度 s_w 等于 111。此外，apDW 波动比 apSW 少了很多，但阻带内没有 −∞ 点，这是因为它不具有零点而致。

以上基于 LMS 准则基窗设计方法的一个前提条件是不增加过渡点，这样得到的窗具有较大的窗口加速度。若侧重改善指标 **A**，则可通过增加不同幅度的过渡点来实现，而幅度的选取也可通过自适应方式实现。此外，算法对 ω_m 的随机选取而得到不同精度下的基窗，如何设置基窗初值和收敛步长是平衡精度和速度关键。

3.3　基于矩形二叉分解的基窗设计

设计 FIR 滤波器时，带宽和阻带最小衰减相互矛盾，窗函数通过在频域与理想滤波器卷积会增大带宽，但可获得较大的最小旁瓣衰减。由 3.1 节介绍的窗函数作用可知，理想窗谱应是一个冲击函数，这样卷积得到的仍是理想滤波器。但冲击谱对应的是无限长时域信号（即不对信号进行截短），这显然不切实际的。高主−旁瓣幅值比和窄主瓣宽度是设计优良窗的两个焦点。基于此提出了“基于矩形窗二叉分解的基窗设计”算法，对各级分解得到的基窗使 apDW 特性在性能 *A* 和 *B* 之间有了较大的灵活选择性。

由式（2-46）并结合定理 2 可知，对于具有线性相位的 apDW 卷积窗 **C** 谱应等于前基窗 **F** 谱与其共轭的乘积，即

$$\boldsymbol{P}_{\mathrm{D}}(\mathrm{e}^{\mathrm{j}\omega})=\boldsymbol{P}_{\mathrm{F}}(\mathrm{e}^{\mathrm{j}\omega})\cdot\boldsymbol{P}_{\mathrm{F}}^{*}(\mathrm{e}^{\mathrm{j}\omega})=|\boldsymbol{P}_{\mathrm{F}}(\omega)|^{2}=\boldsymbol{P}_{\mathrm{D}}(\omega)\qquad(3\text{-}22)$$

式（3-22）说明窗 **C** 谱是恒为正的实函数且幅值等于基窗 **F** 谱值的平方。16 阶三角基窗 **F** 和卷积窗 **C** 谱形状及主瓣占比如图 3-25 所示。

由图 3-25 可以看出，卷积窗 **C** 谱幅频特性改善明显。这是因为基窗 **F** 谱自乘时，主瓣和旁瓣幅值低于 1 的部分都按照二次方算子减弱了，且相对幅值越小减弱幅度越大。其次，共轭谱幅频特性相同使卷积窗 **C** 主瓣宽度和零点个数保

持不变。显然，这种谱相乘次数越多，结果就越接近冲击谱。但对于长度为偶数阶 N 的数字三角窗是由 $N/2$ 阶和 $N/2+1$ 阶两个矩形窗卷积生成，则卷积窗 $\boldsymbol{C}$ 谱的零点会增加 $N/2+1$ 个而过渡带宽度则会降低 $2/(N+2)$。以 Bartlett 为基窗 apDW 的卷积窗 $\boldsymbol{C}_{\mathrm{d}}$ 及其谱 $\boldsymbol{P}_{\mathrm{d}}$ 与基窗 $\boldsymbol{F}$ 之间关系如图 3-26 所示。

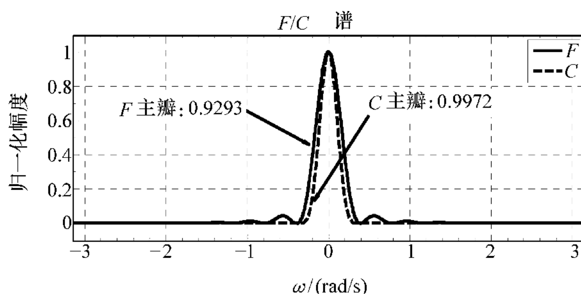


图 3-25　32 阶基窗 $\boldsymbol{F}$ 和卷积窗 $\boldsymbol{C}$ 谱特性

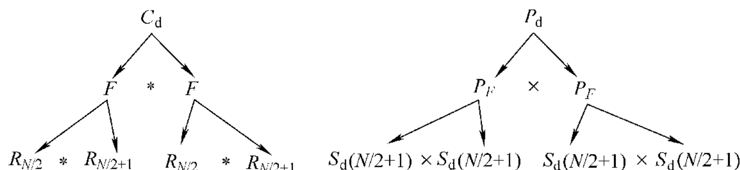


图 3-26　卷积窗 $\boldsymbol{C}$ 和基窗 $\boldsymbol{F}$ 谱对应关系

由图 3-26 可看出，$\boldsymbol{C}_{\mathrm{d}}$ 谱是若干个矩形窗谱乘积，其主瓣宽度应由最小阶窗决定，因此为保持过渡带最低程度放大，每级都应以准等长矩形窗进行分解。将上面过程进一步深化，认为“矩形窗”只是分解过程的中间结果，即它是由低阶矩形窗卷积产生（结果不再是矩形窗）。这样，$\boldsymbol{F}$ 可由若干级二叉分解基窗卷积生成，即

$$\boldsymbol{F}=\boldsymbol{L}_{m_0}^{0}*\boldsymbol{R}_{n_0}^{0};\quad \boldsymbol{L}_{m_0}^{0}=\boldsymbol{L}_{m_1}^{1}*\boldsymbol{R}_{m_2}^{1};\boldsymbol{R}_{n_0}^{0}=\boldsymbol{L}_{n_1}^{1}*\boldsymbol{R}_{n_2}^{1};\quad \cdots\quad \boldsymbol{R}_{3}^{j}=\boldsymbol{R}_{2}^{j+1}*\boldsymbol{R}_{2}^{j+1}\tag{3-23}$$

式中，$\boldsymbol{R}_{n_i}^{j}$ 表示第 j 级第 i 个二叉分解右分支中长度等于 n_i 的窗函数，最终可分解成若干个长度为 2 的矩形窗卷积，只进行 1 级分解的结果 $\boldsymbol{F}$ 就是三角窗。式（3-23）可用图 3-27 来加以说明。

尽管每级分解得到的两子窗长度有多种组合，但为保证卷积窗 $\boldsymbol{C}$ 过渡带宽度最小，应采用准等分方式进行，如 $N=8$ 时的完全分解如图 3-28 所示。

显然，第 k 级父窗与第 $k+1$ 级子窗长度有下面关系：

$$\begin{cases} m_{k+1}=n_{k+1}=(m_k+1)/2 & m_k \text{ 是奇数} \\ m_{k+1}=m_k/2,\ n_{k+1}=m_{k+1}+1 & m_k \text{ 是偶数} \end{cases}\tag{3-24}$$

假设上面二叉树中有 n 个叶片，根结点 $\boldsymbol{F}$ 就是 n 个叶片 $\boldsymbol{R}_2$ 的卷积，所以有

$$N=2n-n+1\Rightarrow n=N-1\tag{3-25}$$

则完全分解得到的卷积窗 $\boldsymbol{C}$ 谱等于

$$\boldsymbol{P}_{\mathrm{d}}(\omega)=\left[\frac{\sin(\omega)}{\sin(\omega/2)}\right]^{2n}=[2\cos(\omega/2)]^{2(N-1)}\tag{3-26}$$

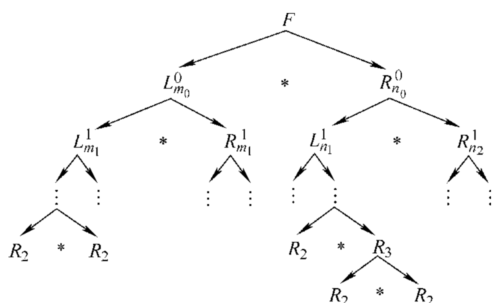


图 3-27　基于矩形窗二叉分解

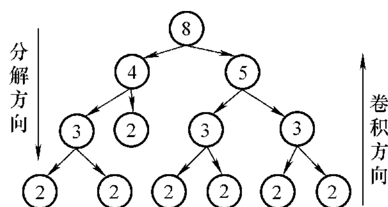


图 3-28　8 阶卷积窗 **C** 的完全分解模式

因为 $\boldsymbol{R}_2$ 的过渡带宽度是 π，所以由式（3-26）完全分解得到的卷积窗 **C** 过渡带很宽，不彻底分解可缓解这一问题。假设 apDW 阶数 $N=2^M$，则二叉分解的最大级数为 M。在第 m 级中（$1\leqslant m\leqslant M-1$）共有 2^m 个矩形，其中最短一个矩形长度为 2^{M-m}，其余 2^m-1 个长度为 $2^{M-m}+1$，则对应的卷积窗 **C** 谱为

$$\boldsymbol{P}_\mathrm{d}=\frac{\left(\sin\dfrac{N\omega}{2^m}\right)^2\left[\sin\left(\dfrac{N}{2^m}+1\right)\omega\right]^{(2^{m+1}-2)}}{\left(\sin\dfrac{\omega}{2}\right)^{2^{m+1}}} \tag{3-27}$$

由其中最小长度矩形决定的过渡带宽度为

$$B=\frac{2\pi}{2^{M-m+1}}=\frac{2^m\pi}{N} \tag{3-28}$$

因此，分解级数每增加一层，相应的 **F** 过渡带宽就增加一倍。如利用二叉等分分解设计 $N=32$ 截止频率为 π/2 的 apDW，完全分解得到各级基窗及特性如图 3-29 所示。

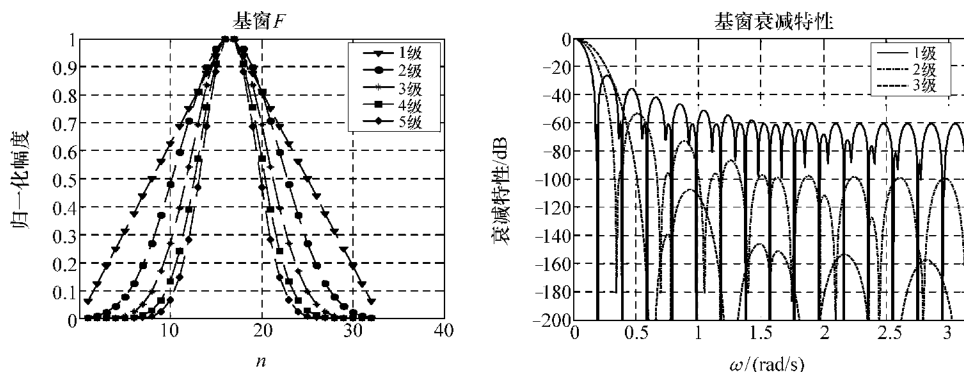


图 3-29　5 级分解基窗 **F** 及特性

由窗谱衰减特性可看出，随着分解层数增加，最大旁瓣峰值急速下降，第 4 级别达到 −209.56dB，第 5 级则为 −358.14dB。各级别基窗 **F** 特性见表 3-3。

表 3-3　各级别基窗 F 指标

	主瓣宽度	B（$\Delta\omega$）	A/dB	D/(dB/oct)
1 级	π/16	0.62	-26.62	-26.46
2 级	1.78π/16	0.83	-53.56	-70.77
3 级	3.18π/16	1.06	-138.92	-177.32
4 级	5.34π/16	1.26	-209.56	-86.04
5 级	7.97π/16	1.35	-358.14	×

第 5 级基窗 F 没有旁瓣，所以无法给出它的第一旁瓣衰减 A 和倍频衰减 D。比较表 3-2 与表 3-3 可以看出，二叉分解得到的基窗具有很大的波峰衰减率和较窄 -3dB 带宽，是新类型窗函数，其缺点是高级分解时过渡带较宽，但通过加大 N 可缓解。

以各基窗 F 建立的滤波器特性如图 3-30 所示。

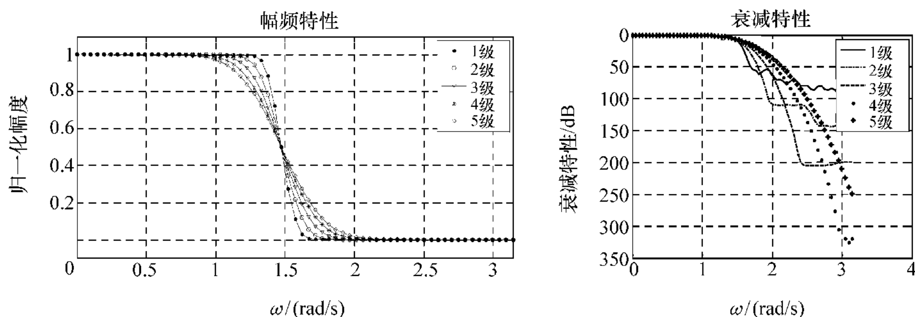


图 3-30　基于各级基窗 F 的 apDW 特性

由实验结果可以看出，基于二叉分解法得到基窗的 apDW 具有十分理想的衰减特性。等分二叉分解层次越高，衰减特性越好，但过渡带较宽。图 3-30 中，第 3 级分解曲线旁瓣波纹十分理想。实现上述结果的 MATLAB 代码如下（只列出二叉分解函数 biTree，滤波器特性曲线 genapDW 及画图函数 todraw 省略）：

```
function biDiv(N)
if nargin = =0
   N =32;wp =pi/2;
end
wins =biTree(N);
aps =genapDW(N,wp,wins);
todraw(wins,aps);

function wins =biTree(N)
n =1;yn =0;n2 =0;
L =(N +mod(N,2))/2 +1 -mod(N,2);
```

```
R=(N+mod(N,2))/2;M=log2(N);
if(N-2^floor(M))>1
   M=floor(M)+1;
else
   M=floor(M);
end
layer=cell(1,M);layer{1}=[L,R];
len=length(layer{1});
notover=sum(layer{n}(:)>2);
while notover
   for i=1:len
      if yn==1
         x=layer{n-1}(i);
      else
         x=layer{n}(i);
      end
      if(x>2)
         if(yn==0)
            n=n+1;yn=1;
         end
         L=(x+mod(x,2))/2+1-mod(x,2);
         R=(x+mod(x,2))/2;
         layer{n}=[layer{n},L,R];
      else
         n2=n2+1;
      end
   end
   yn=0;len=length(layer{n});
   notover=sum(layer{n}(:)>2);
end

for i=1:n2
   layer{M}=[layer{M},2];
end
for i=1:M
   len=length(layer{i});
   RL=ones(layer{i}(1),1);
   RR=ones(layer{i}(2),1);
   temp=conv(RL,RR);
   if len>2
      for k=3:len
```

```
            temp = conv(temp,ones(layer{i}(k),1));
        end
    else
        out(:,i) = temp;
    end
    out(:,i) = temp;
end
for i =1:M
    out(:,i) = out(:,i)/max(out(:,i));
end
wins.y = out;
wins.x = [1:N];
wins.z = 'w';
```

3.4 最小二乘基窗设计算法

由第 2 章内容可知，对于满足线性条件 ap 系统传输特性可表示为式（2-51），令

$$S(n\omega)=\begin{cases}h(0) & n=0\\ 2h(n)\cos(n\omega) & n\in[1,N-1]\end{cases} \tag{3-29}$$

重写传输特性如下

$$H(\mathrm{e}^{\mathrm{j}\omega})=\sum_{n=0}^{N-1}c_nS(n\omega) \tag{3-30}$$

假设频率取样序列为 $\boldsymbol{H}$，则传输特性的频率偏差距离为

$$\begin{aligned}ERR &= \left\|\boldsymbol{H}-H(\mathrm{e}^{\mathrm{j}\omega})\big|_{\omega_k}\right\|_2=\left\|H(k)-\sum_{n=0}^{N-1}c_nS(n\cdot\omega_k)\right\|_2\\ &=\sum_{k=0}^{N-1}\left[H(k)-\sum_{n=0}^{N-1}c_nS(n\cdot\omega_k)\right]^2\end{aligned} \tag{3-31}$$

假设 N 为奇数（偶数同法可证），根据卷积窗中心对称性整理式（3-31）得

$$\begin{aligned}ERR &= \sum_{k=0}^{N-1}\left\{H(k)-\left[c_0h_0+\sum_{n=1}^{(N-1)/2}c_n\cos(n\cdot\omega_k)+\sum_{n=(N+1)/2}^{N-1}c_n\cos(n\cdot\omega_k)\right]\right\}^2\\ &=\sum_{k=0}^{N-1}\left\{H(k)-\left[c_0h_0+\sum_{n=1}^{(N-1)/2}c_n\cos(n\cdot\omega_k)+\sum_{n=0}^{(N-1)/2}c_{N-n}\cos((N-n)\cdot\omega_k)\right]\right\}^2\\ &=\sum_{k=0}^{N-1}\left\{H(k)-\left[c_0h_0+\sum_{n=1}^{(N-1)/2}c_n\left(\cos(n\omega_k)+\cos((N-n)\omega_k)\right)\right]\right\}^2\end{aligned} \tag{3-32}$$

式中，$\omega_k = 2k\pi/N$。再令

$$T(n\omega) = \begin{cases} h(0) & n = 0 \\ 4h(n)\cos(n\omega) & n \in [1, (N-1)/2] \end{cases} \tag{3-33}$$

则式（3-32）表示的频采误差又可表示为

$$ERR = \sum_{k=0}^{N-1} \left\{ H(k) - \sum_{n=0}^{(N-1)/2} c_n T(n \cdot \omega_k) \right\}^2 \tag{3-34}$$

由于 $\boldsymbol{H}$ 具有对称性，所以式（3-34）进一步简化如下：

$$\left[H(0) - \sum_{n=0}^{(N-1)/2} c_n T(n \cdot \omega_0) \right]^2 + 2\sum_{k=1}^{(N-1)/2} \left\{ H(k) - \sum_{n=0}^{(N-1)/2} c_n T(n \cdot \omega_k) \right\}^2 \tag{3-35}$$

式（3-35）是卷积窗 $\boldsymbol{C}$ 的函数，为取得最小值，应满足：

$$\begin{aligned} \frac{\partial ERR}{\partial c_n} = 0 = &-2T(m\omega_0)\left[H(0) - \sum_{n=0}^{(N-1)/2} c_n T(n\omega_0) \right] \\ &-4\sum_{k=1}^{(N-1)/2} \left[H(k) - \sum_{n=0}^{(N-1)/2} c_n T(n\omega_k) \right] T(m\omega_k) \end{aligned} \tag{3-36}$$

即误差对于任意卷积窗元素 c_n 的偏导数恒为零，整理式（3-36）得：

$$\begin{aligned} \frac{\partial ERR}{\partial c_n} &= T(m\omega_0) \sum_{n=0}^{(N-1)/2} c_n T(n\omega_0) + 2\sum_{k=1}^{(N-1)/2} \sum_{n=0}^{(N-1)/2} c_n T(n\omega_k) T(m\omega_k) \\ &= T(m\omega_0) H(0) + 2\sum_{k=1}^{(N-1)/2} H(k) T(m\omega_k) m \in \left[0, \frac{(N-1)}{2}\right] \end{aligned} \tag{3-37}$$

定义如下矩阵：

$$\boldsymbol{P} = \begin{bmatrix} COH(0) & COH(\omega_0) & \cdots & COH\left(\frac{N-1}{2}\omega_0\right) \\ COH(0) & COH(\omega_1) & \cdots & COH\left(\frac{N-1}{2}\omega_1\right) \\ \vdots & \vdots & \vdots & \vdots \\ COH(0) & COH(\omega_{(N-1)/2}) & \cdots & COH\left(\frac{N-1}{2}\omega_{(N-1)/2}\right) \end{bmatrix}_{\frac{N+1}{2} \times \frac{N+1}{2}}$$

$$\boldsymbol{V} = \begin{bmatrix} COH(0) & 2COH(0) & \cdots & 2COH(0) \\ COH(\omega_0) & 2COH(\omega_1) & \cdots & 2COH(\omega_{(N-1)/2}) \\ \vdots & \vdots & \vdots & \vdots \\ COH\left(\frac{N-1}{2}\omega_0\right) & 2COH\left(\frac{N-1}{2}\omega_1\right) & \cdots & 2COH\left(\frac{N-1}{2}\omega_{(N-1)/2}\right) \end{bmatrix}_{\frac{N+1}{2} \times \frac{N+1}{2}}$$

$$\boldsymbol{C} = \begin{bmatrix} c_0 & c_1 & \cdots & c_{\frac{N-3}{2}} & c_{\frac{N-1}{2}} \end{bmatrix}^{\mathrm{T}}_{\frac{N+1}{2} \times 1}, \quad \boldsymbol{H} = \begin{bmatrix} H_0 & H_1 & \cdots & H_{\frac{N-3}{2}} & H_{\frac{N-1}{2}} \end{bmatrix}^{\mathrm{T}}_{\frac{N+1}{2} \times 1}$$

则式（3-37）的矩阵表达如下：

$$\boldsymbol{VPC} = \boldsymbol{VH} \tag{3-38}$$

从而求得卷积窗 $\boldsymbol{C}$ 为

$$\boldsymbol{C} = (\boldsymbol{VP})^{-1}\boldsymbol{VH} \tag{3-39}$$

假设 $N=17$ 的低通滤波器上截止频率为 $\pi/2$，则传输特性序列 $\boldsymbol{H}=$［1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1］，其频率取样 $\omega_k = 2k\pi/17$。求得单位响应序列 $\boldsymbol{h}=$［0.5294 0.3188 −0.0299 −0.1075 0.0315 0.0660 −0.0346 −0.0488 0.0398 0.0398 −0.0488 −0.0346 0.0660 0.0315 −0.1075 −0.0299 0.3188］，则分别构造矩阵 $\boldsymbol{P}$、$\boldsymbol{V}$、$\boldsymbol{C}$、$\boldsymbol{H}$ 代入式（3-39）求得窗 $\boldsymbol{C}=$［0.50 0.50 0.50 0.50 0.50 0.50 0.50 0.50 1.00 0.50 0.50］，如果以 $\boldsymbol{C}$ 为基窗，则卷积窗图形及对应的系统曲线如图 3-31 所示。

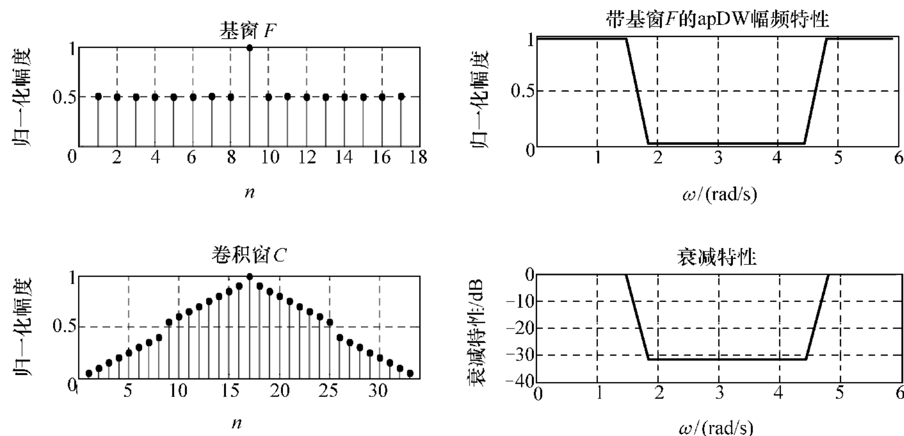


图 3-31 基于最小二乘的基窗 $\boldsymbol{F}$ 卷积窗 $\boldsymbol{C}$ 及滤波器特性

实现上述结果的 MATLAB 代码如下（画图部分省略）：

```
function ms2FW(N)
ifnargin = =0
   N =17;wz =pi/2;
end
nL =(N -1)/2;n1 =floor(wz/pi* nL);
H1 =zeros(1,nL);H1(1:n1) =1;
H =[1 H1 fliplr(H1)];HH =H(1:nL +1);
h =ifft(H);hh =h(1:nL +1);

w =2* pi/N* [0:nL];
P =zeros(nL +1,nL +1);
P1(1:nL +1) =h(1);
```

```
wij = w'* [1:nL];cwij = cos(wij);
harray = ones(nL +1,1)* h(2:nL +1);
P2 =4* harray.* cwij;P = [P1'P2];PP = P';
eff =2* ones(nL +1,nL +1);eff(:,1) =1;
V = PP.* eff;Y = V* HH';XX = inv(V* P)* Y;
XX = XX';X = [fliplr(XX(2:end))XX];
CNc = conv(X,X);CNc = CNc/max(CNc);CN = CNc(N:2* N -1);
ERR =2* sum(abs(HH - XX* P).^2);
```

卷积窗 ***C*** 性能参数 $B=0.2165$，$A=-14\text{dB}$，$D=-10.15\text{dB}$。虽然 A 和 D 较低，但过渡带宽度降低很多，窗口速度 s_w 达到了 64.67，加速度 a_w 达到了 46.89。对比图 3-16 七种基窗可看出，利用最小二乘得到的基窗综合性能最好。此外，基于此结果基窗 ***F*** 的滤波器具有最小的均方误差。由图 3-31 特性曲线可看出，虽然通带幅度未达到 1 且阻带幅度触底到 0，但系统整体误差为 0.2117，比已知最优三角窗仍降低了 2%。

3.5　小结

本章首先分析窗函数在滤波器设计中的作用，通过实验证明了增加阶数 N 并不是改善吉布斯效应的有效途径。其次，对 7 种常用基窗的特性进行了分析比较，提出窗口速度和窗口加速度的概念。第三，分析了基于三角窗的 apDW 频率采样误差与阶数 N 的变化关系，总结出当阶数大于 80 时的误差收敛稳定的结论。最后，讲解了以 ap 为模型的三种窗设计算法即 LMS 算法、二叉分解法和最小二乘法。所有的结论均通过 MATLAB 编程进行了实验验证。

第 4 章 正交域全相位变换

4.1 正交变换域

在数字滤波和数字信号处理系统中，广泛使用的运算是卷积和相关运算。若直接应用时域法进行处理，计算量将随着抽样点数的二次方而增加。对于很大的点数，就使计算机的负担加重，计算也很费时，难以达到实时处理的要求。因此，一般都采用变换方法，即把输入信号先进行变换，处理后再进行反变换，然后恢复为时域信号。用计算机对变换后的信号进行处理，往往比直接处理更加方便，还能大大减少运算次数而提高处理速度。

在信号处理中，最常用的变换方法是 DFT。1965 年 Cooley – Tukey 提出的快速傅里叶变换（FFT）方法，可使原来需要 N^2 次运算的算法变为只需 $N/2\log_2 N$ 次运算，且被变换的抽样点数 N 越多，FFT 算法节省的计算量就越显著。除了 DFT 之外，还有一些其他的算法，即用不同正交函数系来代替傅里叶变换三角函数系。傅里叶变换基础是由正、余弦函数构成的完备正交函数系，使任何绝对可积的函数都可表示成正、余弦函数，但它进行复数运算就比较费时。离散余弦变换、离散沃尔什变换及离散正弦变换等正交变换则避免了复数运算，运算效率进一步提高。

4.1.1 一维变换的正交基

若均匀采样的输入函数 $f(n)$ 的长度为 L_1，系统单位冲击相应函数 $h(n)$ 长度为 L_2。取 $L \geqslant L_1 + L_2 - 1$，通过补充零值的方式将 $f(n)$ 和 $h(n)$ 扩充为以 L 长度为周期的函数 $f_e(x)$ 和 $h_e(x)$，则系统输出 $g_e(n)$ 可表示成 $f_e(n)$ 和 $h_e(n)$ 的循环卷积：

$$g_e(n) = \sum_{n=-\infty}^{+\infty} f_e(m) h_e((n-m))_L \tag{4-1}$$

显然 $g_e(n)$ 是长度为 L 的周期函数，其主值区间输出可用矩阵形式表示如下：

$$\boldsymbol{G}_e = \boldsymbol{H}_e \cdot \boldsymbol{F}_e \tag{4-2}$$

式中，$\boldsymbol{F}_{\mathrm{e}} = [f(0), f(1), \cdots, f(L-1)]^{\mathrm{T}}$；$\boldsymbol{G}_{\mathrm{e}} = [g(0), g(1), \cdots, g(L-1)]^{\mathrm{T}}$；$\boldsymbol{H}_{\mathrm{e}}$的定义如下：

$$\boldsymbol{H}_{\mathrm{e}} = \begin{bmatrix} h_{\mathrm{e}}(0) & h_{\mathrm{e}}(L-1) & h_{\mathrm{e}}(L-2) & \cdots & h_{\mathrm{e}}(1) \\ h_{\mathrm{e}}(1) & h_{\mathrm{e}}(0) & h_{\mathrm{e}}(L-1) & \cdots & h_{\mathrm{e}}(2) \\ h_{\mathrm{e}}(2) & h_{\mathrm{e}}(1) & h_{\mathrm{e}}(0) & \cdots & h_{\mathrm{e}}(3) \\ \vdots & \vdots & \vdots & & \vdots \\ h_{\mathrm{e}}(L-1) & h_{\mathrm{e}}(L-2) & h_{\mathrm{e}}(L-3) & \cdots & h_{\mathrm{e}}(0) \end{bmatrix}_{L \times L} \tag{4-3}$$

若$\boldsymbol{H}_{\mathrm{e}}$的任意行、列向量正交，则称其为变换核。一维时域信号按照式（4-2）经过变换核到变换域内任意点的一般表达式为

$$\boldsymbol{T}(n) = \frac{1}{N}\sum_{m=0}^{M-1}\boldsymbol{F}(m)\boldsymbol{H}(n, m) \Leftrightarrow \boldsymbol{T} = \frac{1}{N}\boldsymbol{H} \cdot \boldsymbol{F} \tag{4-4}$$

式中，矩阵 $\boldsymbol{H}$ 为酉算子，满足 $\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H} = \boldsymbol{H}\boldsymbol{H}^{\mathrm{T}} = N\boldsymbol{I}$。对式（4-4）做逆变换就可得

$$\boldsymbol{F} = \boldsymbol{H}^{\mathrm{T}}\boldsymbol{T} \tag{4-5}$$

将 $\boldsymbol{H}$ 用列向量表示如下：

$$\begin{aligned} &\boldsymbol{H} = [\boldsymbol{h}_0 \quad \boldsymbol{h}_1 \quad \boldsymbol{h}_2 \quad \cdots \quad \boldsymbol{h}_{N-1}]^{\mathrm{T}}，其中 \\ &\boldsymbol{h}_i = [h(i, 0) \quad h(i, 1) \quad h(i, 2) \quad \cdots \quad h(i, N-1)]^{\mathrm{T}} \end{aligned} \tag{4-6}$$

将向量 $\boldsymbol{T}$ 写成如下求和方式，有

$$\boldsymbol{T} = \begin{bmatrix} t_0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ t_1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ t_2 \\ \vdots \\ 0 \end{bmatrix} + \cdots + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ t_{N-1} \end{bmatrix} = \sum_{i=0}^{N-1}\boldsymbol{t}_i \tag{4-7}$$

把式（4-6）和式（4-7）代入到式（4-5）后并整理得：

$$\boldsymbol{F} = \sum_{i=0}^{N-1}\boldsymbol{h}_i^{\mathrm{T}}\sum_{j=0}^{N-1}\boldsymbol{t}_j \tag{4-8}$$

由变换核的各列向量 $\boldsymbol{h}_i$构成相互正交的基本信号。式（4-8）表示以变换域系数 $\boldsymbol{t}_j$作加权的基本信号之和就可构建出原始信号 $\boldsymbol{F}$，$\boldsymbol{h}_i$称为变换正交基，而加权系数的含义与选择的正交基相关。

4.1.2　沃尔什正交基

对于长度为 N 的一维信号 $\boldsymbol{X}$，沃尔什变换定义 $\boldsymbol{T}_{\mathrm{W}}$为

$$\boldsymbol{T}_{\mathrm{W}} = \boldsymbol{Wal}_{\mathrm{N}} \cdot \boldsymbol{X} \tag{4-9}$$

沃尔什正交基元素由下列沃尔什函数生成：

$$\mathrm{Wal}(k,t)=\prod_{i=0}^{r-1}\mathrm{sgn}[\cos(k_i\cdot 2^i\cdot\pi\cdot t)],\quad k=\sum_{i=0}^{r-1}k_i 2^i,\quad k_i\in\{0,1\}\tag{4-10}$$

式中，sgn（·）是符号函数，即对于正数为 +1，对于负数为 −1；k_i是 k 用二进制表示的第 i 位数字；t 为 0 到 $N-1$ 归一化表示，即 m/N（$m=0, 1, \cdots, N-1$）。例如对于 $N=4$，$r=2$，$k=0\sim3$ 二进制表示 k_0k_1分别为00/01/10/11 时，沃尔什基底函数组成见表 4-1。

表 4-1　$N=4$ 的沃尔什函数

$N=4$ 的沃尔什函数（列率表示式）	$t=m/N(0, 1/4, 2/4, 3/4)$
$\mathrm{Wal}(0,t)=\mathrm{sgn}[\cos(0\cdot\pi\cdot t)]$	1, 1, 1, 1
$\mathrm{Wal}(1,t)=\mathrm{sgn}[\cos(1\cdot\pi\cdot t)]$	1, 1, −1, −1
$\mathrm{Wal}(2,t)=\mathrm{sgn}[\cos(1\cdot 2^1\pi\cdot t)]\cdot\mathrm{sgn}[\cos(0\cdot 2^0\pi\cdot t)]=\mathrm{sgn}[\cos(2\pi\cdot t)]$	1, −1, −1, 1
$\mathrm{Wal}(3,t)=\mathrm{sgn}[\cos(1\cdot 2^1\pi\cdot t)]\cdot\mathrm{sgn}[\cos(1\cdot 2^0\pi\cdot t)]=\mathrm{Wal}(2,t)\cdot\mathrm{Wal}(1,t)$	1, −1, 1, −1

MATLAB 实现沃尔什函数的代码如下：

```
function walFun(N)
if nargin = =0
   N =4;M =N* 3;wid =M* N;dis =4;
end
walf =zeros(N,N);
for k =0:N -1
   for tn =0:N -1
       t =tn/N;bn =dec2bin(k);len_bn =length(bn);ini =1;
       for i =0:len_bn -1
           ki =bin2dec(bn(len_bn -i));
           theta =ki* 2^i* pi* t;yn =theta* 2/pi;
           ini =ini* sign(cos(theta));
           if(yn -floor(yn) = =0)&&(yn > =1)&& mod(yn,2)
               ini = -ini;
           end
       end
       walf(k +1,tn +1) =ini;
   end
end
wf =zeros(N,wid);xtklb ={};ytk =[ -1 1];ytklb ={};
for i =1:N
   wf(:,(i -1)* M +1:i* M) =repmat(walf(:,i),1,M) +...
       repmat([dis* (N -1): -dis:0]',1,M);
end
```

```
h =plot([1:wid],wf,'k','LineWidth',3);
denormstr =num2str(N);denormstr =strcat('/',denormstr);
for k =0:N -1
    xtklb ={xtklb{:},strcat(num2str(k),denormstr)};
    if k <N -1
        ytk =[ytk,[ytk(end -1),ytk(end)] +dis];
        ytklb ={ytklb{:},' -1','1'};
    end
end
ytks =[ytk(1:end -2),ytk(end)];len_ytk =length(ytks);
ytklb ={ytklb{:},'1'};
set(gca,'FontSize',16);
set(gca,'XTick',[M/2:M:wid -M/2],'XTickLabel',xtklb);
set(gca,'YTick',ytks,'YTickLabel',ytklb);grid on;
xlabel('t');ylabel('amp');title(strcat(num2str(N),'order Walsh base'));
xcor =wid -M/2;strn1 ='Wal(';strn3 =',t)';
for k =1:N
    ycor =(N +1 -k)* dis -dis;
    strn2 =strcat(num2str(k -1),strn3);
    text(xcor,ycor,strcat(strn1,strn2),'FontSize',16);
end
```

运行结果如图 4-1 所示。

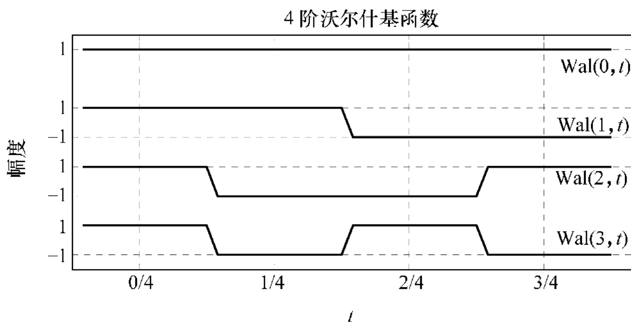


图 4-1　4 阶沃尔什基函数

对于 N 阶沃尔什变换基，定义在 N 点间隔内过零次数的一半定义为列率（sequency）。由表 4-1 可看出，按列率排列的沃尔什基序号 k 是过零次数，故对应列率 $k/2$；反之，沃尔什基的沃尔什变换是仅在列率为 $k/2$ 时非零的 N 向量。对于沃尔什变换具有以下重要性质。

性质 1：两个不同列率的沃尔什基形成完备正交系，即基相乘产生列率搬移。

证明：不同频率 f_m 和 f_n 的正弦函数相乘会出现 $f_m \pm f_n$ 频率分量，即频率会发生搬移。对于列率分别为 m 和 n 的沃尔什基相乘过程如下：

$$\begin{aligned}\mathrm{Wal}(m,t)\cdot\mathrm{Wal}(n,t) &= \prod_{i=0}^{r_m-1}\mathrm{sgn}[\cos(m_i\cdot 2^i\pi\cdot t)]\prod_{j=0}^{r_n-1}\mathrm{sgn}[\cos(n_j\cdot 2^j\pi\cdot t)]\\ &= \prod_{i=0}^{r-1}\mathrm{sgn}[\cos(m_i\cdot 2^i\pi\cdot t)]\mathrm{sgn}[\cos(n_i\cdot 2^i\pi\cdot t)],\\ r &= \max\{r_m,r_n\}\end{aligned} \tag{4-11}$$

对于 m_i，$n_i\in\{0,1\}$，区分相等和不等两种情况分析易知式（4-11）等于：

$$\mathrm{Wal}(m,t)\cdot\mathrm{Wal}(n,t) = \prod_{i=0}^{r-1}\mathrm{sgn}\{\cos[(m_i\otimes n_i)\cdot 2^i\pi\cdot t]\} = \mathrm{Wal}(m\otimes n,t) \tag{4-12}$$

式（4-12）说明两沃尔什基函数相乘得到列率搬移的沃尔什基且其列率等于原两列率的模 2 异或。表 4-2 列出 $N=8$ 时的搬移规律。

表 4-2　$N=8$ 列率搬移

n \ m	0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6	7
1	1	0	3	2	5	4	7	6
2	2	3	0	1	6	7	4	5
3	3	2	1	0	7	6	5	4
4	4	5	6	7	0	1	2	3
5	5	4	7	6	1	0	3	2
6	6	7	4	5	2	3	0	1
7	7	6	5	4	3	2	1	0

性质 2：信号并元移位 s 的沃尔什变换等于信号沃尔什变换与列率 s 和沃尔什基乘积。

证明：对于沃尔什基坐标 k 和 t（整数）取值线形无关均为 $[0,N-1]$，所以由式（4-10）定义的沃尔什基矩阵具有对称性，即

$$\mathrm{Wal}(k,t)=\mathrm{Wal}(t,k) \tag{4-13}$$

记信号 $[f(x)]=[f(0),f(1),f(2),f(3)]$，定义 $[f(x\otimes s)]$ 为列率并元移位 s 位后的序列，用 $[f(x)_s]$ 来表示。显然，$x\otimes s$ 的意义为信号 $[f(x)]$ 中各元素序号与 s 异或得到的重排序即列率搬移。设 $[f(x)]$ 和 $[f(x)_s]$ 的沃尔什变换分别为 $[F(k)]$ 和 $[F(k)_s]$，即

$$\begin{aligned}F(k)_s &= \sum_{x=0}^{N-1}f(x)_s\mathrm{Wal}(k,x) = \sum_{x=0}^{N-1}f(x\otimes s)\mathrm{Wal}(k,x)\\ &= \sum_{x'=0}^{N-1}f(x')\mathrm{Wal}(k,x'\otimes s),\quad x'=x\otimes s\\ &= \sum_{x'=0}^{N-1}f(x')\mathrm{Wal}(k,x')\mathrm{Wal}(k,s)\\ &= F(k)\cdot\mathrm{Wal}(k,s)\end{aligned} \tag{4-14}$$

如 $[f(x)]=[1, 2, -1, 3]$，则 $[F(k)]=[5, 1, 3, -5]/4$；$[f(x)_2]=[-1, 3, 1, 2]$，求得对应的沃尔什变换 $[F(k)_2]=[5, -1, -3, -5]/4=[F(k)]\cdot \mathrm{Wal}(k, 2)$，证明了结论的正确性。

性质 3：信号循环并元卷积的沃尔什变换等于信号沃尔什变换的乘积。

证明：信号 $f(x)g(x)$ 的循环并元卷积定义如下：

$$C(s) = \frac{1}{N}\sum_{x=0}^{N-1} f(x)g(x\otimes s) \tag{4-15}$$

$C(s)$ 的沃尔什变换如下：

$$\begin{aligned}
C(k) &= \frac{1}{N^2}\sum_{s=0}^{N-1} C(s)\mathrm{Wal}(k, s) = \frac{1}{N^2}\sum_{s=0}^{N-1}\sum_{x=0}^{N-1} f(x)g(x\otimes s)\mathrm{Wal}(k, s), \quad x' = x\otimes s \\
&= \frac{1}{N^2}\sum_{x=0}^{N-1} f(x)\sum_{x'=0}^{N-1} g(x')\mathrm{Wal}(k, x'\otimes x) \\
&= \frac{1}{N}\sum_{x=0}^{N-1} f(x)\mathrm{Wal}(k, x)\cdot\frac{1}{N}\sum_{x'=0}^{N-1} g(x')\mathrm{Wal}(k, x') \\
&= F(k)\cdot G(k)
\end{aligned} \tag{4-16}$$

沃尔什反变换如下：

$$\boldsymbol{X} = \frac{1}{N}\boldsymbol{Wal}_{\mathrm{N}}\cdot \boldsymbol{T}_{\mathrm{W}} \tag{4-17}$$

沃尔什变换主要应用在图像压缩、语音处理、滤波和功率谱分析方面。

4.1.3　余弦正交基

设 $f(x)$，$x=0, 1, 2, \cdots, N-1$，为离散序列，按照下式延拓成偶对称序列 $f_s(x)$：

$$f_s(x+1/2)=f(x) \text{ 且 } f(x)=f(-x-1) \tag{4-18}$$

式中，$x\in[-N, N-1]$。显然 f_s 关于 $x=-1/2$ 偶对称，如图 4-2 所示。

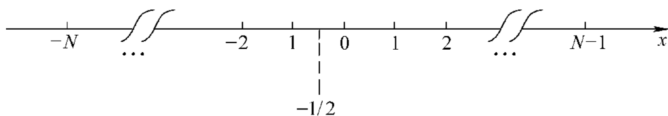


图 4-2　关于 $x=-1/2$ 为中心的偶对称序列

令 $x'=x+1/2$，于是 $f_s(x')$ 是以 $x'=0$ 为中心的偶对称函数，其傅里叶变换为

$$\begin{aligned}
F_s(u) &= \sqrt{\frac{1}{2M}}\sum_{x=-N}^{N-1} f_s(x')\exp[-\mathrm{j}2u\pi(x+1/2)/2N] \\
&= \sqrt{\frac{2}{M}}\sum_{x=0}^{N-1} f_s(x')\cos[(2x+1)u\pi/2N],
\end{aligned} \tag{4-19}$$

所以 $$f_s(0)=0$$

根据式（4-18）知 $f_s(x')$ 和 $f(x)$ 的关系如下：

$$f_s(-x+1/2)=f(-x)=f(x-1)=f_s(x-1/2),\quad x\in[0,N] \tag{4-20}$$

结合式（4-19）和式（4-20）可得 $f_s(x')$ 的傅里叶变换为

$$F_S(u)=\sqrt{\frac{2}{N}}\sum_{x=0}^{N-1}f(x)\cos[(2x+1)u\pi/2N] \tag{4-21}$$

由式（4-21）确定的变换称为离散余弦变换（DCT）。对于长度为 N 的一维信号 $\boldsymbol{X}$，余弦变换定义 $\boldsymbol{T}_C$ 为

$$\boldsymbol{T}_C=\boldsymbol{C}_N\cdot\boldsymbol{X} \tag{4-22}$$

正弦正交基元素由下列函数生成：

$$\boldsymbol{C}_N=[c_{i,j}]_{N\times N},\quad c_{i,j}=\begin{cases}1/\sqrt{N} & i=0\\ \sqrt{\frac{2}{N}}\cos\left(i\pi\cdot\frac{2j+1}{2N}\right) & i=1,2,\cdots,N-1;\ j=0,1,\cdots,N-1\end{cases} \tag{4-23}$$

8 阶正弦变换基函数如图 4-3 所示。

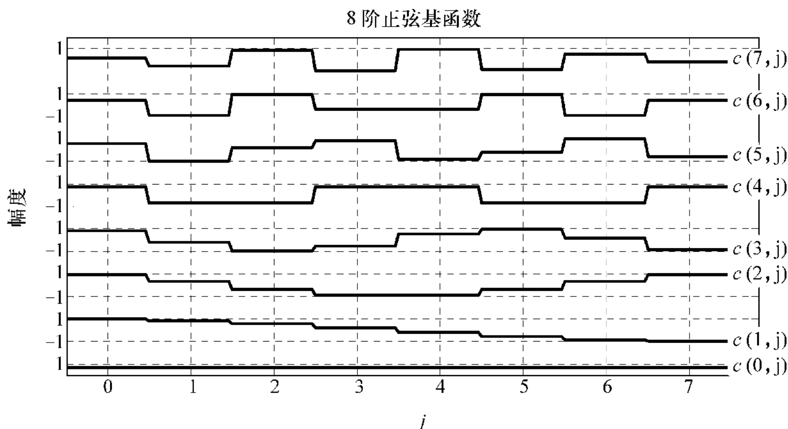


图 4-3 8 阶正弦变换基函数

实现上述结果的 MATLAB 代码如下（省略画图部分）：

```
function cosij =cosFun(N)
if nargin = =0
   N =8;M =N* 3;wid =M* N;dis =4;
end
i =[0:N -1]';j =[0:N -1];
cosij =cos(i* pi* (2* j +1)/2/N);
cosij(1,:) =1/sqrt(2);
y =[];
```

```
fori =1:N
    c =repmat(cosij(:,i),1,M);
    y =cat(2,y,c);
end
ydis =[0:dis:(N -1)* dis]';yds =repmat(ydis,1,wid);
y =y +yds;plot([0:wid -1],y,'k','LineWidth',3);
```

对于 N 阶正弦变换基 C_{N} 满足：

$$\boldsymbol{C}_{\mathrm{N}}\boldsymbol{C}_{\mathrm{N}}^{\mathrm{T}} = \left[\sum_{k=0}^{N-1} c_{i,k}c_{j,k}\right]_{N\times N} = (\boldsymbol{C}_{\mathrm{N}}\boldsymbol{C}_{\mathrm{N}}^{\mathrm{T}})^{\mathrm{T}} = \boldsymbol{I}_{\mathrm{N}} \tag{4-24}$$

式中，$\boldsymbol{I}$ 是 N 阶单位矩阵。证明如下：

证明：若 $i=j=0$，显然有：

$$\sum_{k=0}^{N-1} c_{0,k}c_{0,k} = \sum_{k=0}^{N-1}\frac{1}{N} = 1 \tag{4-25}$$

一般情况下，结合（4-23）得：

$$\begin{aligned}\sum_{k=0}^{N-1} c_{i,k}c_{j,k} &= \frac{2}{N}\sum_{k=0}^{N-1}\cos\left(i\pi\frac{2k+1}{2N}\right)\cos\left(j\pi\frac{2k+1}{2N}\right)\\ &= \frac{1}{N}\sum_{k=0}^{N-1}\left\{\cos\left[(i+j)\pi\frac{2k+1}{2N}\right]+\cos\left[(i-j)\pi\frac{2k+1}{2N}\right]\right\}\\ &= \frac{1}{N}(A+B)\end{aligned} \tag{4-26}$$

当 $i\neq j\neq 0$ 时，令 $n=(i+j)/2N$，显然有 $0<n<1$。式（4-26）中元素 A 为

$$\begin{aligned}A &= \frac{1}{\sin(n\pi)}\sum_{k=0}^{N-1}\{\sin(n\pi)\cos[(2k+1)n\pi]\}\\ &= \frac{1}{2\sin(n\pi)}\sum_{k=0}^{N-1}\{\sin[(2k+2)n\pi]-\sin[2kn\pi]\}\\ &= \frac{\sin(2nN\pi)}{2\sin(n\pi)}\\ &= 0\end{aligned} \tag{4-27}$$

同理可证，$B=0$；当 $i=j\neq 0$ 时，同法可知 A 等于 0，但显然 $B=N$；当 $i=0$ 或 $j=0$ 时，结论同 $i\neq j\neq 0$ 情形。综合以上，可知：

$$\sum_{k=0}^{N-1} c_{i,k}c_{j,k} = \begin{cases}1 & i=j\\ 0 & i\neq j\end{cases} \tag{4-28}$$

结合式（4-22）和式（4-24），可知余弦反变换为

$$\boldsymbol{X} = \boldsymbol{C}_{\mathrm{N}}^{\mathrm{T}}\cdot\boldsymbol{T}_{\mathrm{C}} \tag{4-29}$$

需要指出的是，由于余弦变换基元素均为小数，在运算上较复杂。可通过近似为整数后变换，称为整数 DCT，在 H.264 及 AVS 标准中被使用。对于 4 阶 DCT，余弦变换基等于：

$$\boldsymbol{C}_4 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 & 1 & 1 \\ c & s & -s & -c \\ 1 & -1 & -1 & 1 \\ s & -c & c & -s \end{bmatrix} \tag{4-30}$$

式（4-30）中，$c=\sqrt{2}\cos(\pi/8)$ 和 $s=\sin(\pi/8)$。近似取整的方式可以通过基乘上 α 后作某种规则的取整，也可通过按式（4-29）进行部分运算后步骤取整。不论采用哪种方式都会引入误差，如何分析误差分布规律及有效降低误差是值得研究的问题。余弦变换主要应用在数据内插、图像压缩方面，在 MPEG-2、H.264、AVS 等音视频压缩标准中得到了广泛应用。

4.1.4 其他正交变换

4.1.4.1 哈达玛变换

由上面分析可知，离散沃尔什变换是对［0，1］区间上的连续沃尔什函数进行等间隔取样的结果，因为沃尔什函数是一组完备正交函数集，故可用于正交变换。从本质上说，哈达玛变换与沃尔什变换是相同的，它们的区别在于当取样数 N 是 2 的整数幂时，这两类变换基仅仅是行（列）的顺序不同；当 N 不是 2 的整数幂时，沃尔什变换可用于任意正整数，而哈达玛变换仅能用到 $N=200$。在数字信号处理中，为了可进行快速运算，取样数通常取 2 的整数次幂，因而沃尔什变换和哈达玛变换这两种称呼常被混用。有时，这两类变换被统称为沃尔什-哈达玛变换。哈达玛变换基具有简单的递推关系，给计算带来很大方便。一维哈达玛正变换核由下列公式给出：

$$a(i,j)=(1/\sqrt{N})(-1)^{p(i,j)},\quad i,j=0,1,2,\cdots,N-1 \tag{4-31}$$

式中

$$p(i,j)=\sum_{n=0}^{N-1}k_n(i)k_n(j),\quad i,j=0,1,2,\cdots,N-1 \tag{4-32}$$

需注意的是，式（4-32）中符号 $\sum$ 表示模 2 加法，$k_n(i)$ 表示数 z 的二进制表示中第 i 个比特值。按照式（4-31）生成哈达玛矩阵的 MATLAB 代码如下：

```
function hadbt = hadFun(N)
N = 4;M = N* 3;wid = M* N;dis = 4;
hadb = zeros(N,N);
for i = 0:N -1
    for j = 0:N -1
        ib = dec2bin(i);jb = dec2bin(j);len_ib = length(ib);
```

```
        len_jb = length(jb);maxl = max(len_ib,len_jb);
        if len_ib > len_jb
            jb = strcat(num2str(zeros(1,maxl - len_jb)),jb);
        else
            ib = strcat(num2str(zeros(1,maxl - len_ib)),ib);
        end
        p = 0;
        for k = 0:maxl - 1
            ki = bin2dec(ib(maxl - k));kj = bin2dec(jb(maxl - k));
            p = xor(p,ki* kj);
        end
        hadb(i +1,j +1) = ( -1)^p;
    end
end
```

$N=4$ 时运行结果如图 4-4 所示。MATLAB 定义的哈达玛内联函数是 hardmard(n)。当 $N=2^n$（n 为整数）的哈达玛变换基可通过下面递推公式求得：

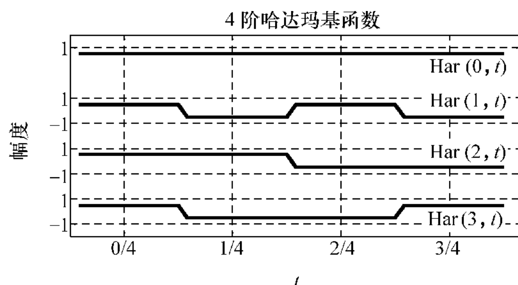


图 4-4　4 阶哈达玛基函数

$$\boldsymbol{H}_{2N}=\frac{1}{\sqrt{2}}\begin{bmatrix}\boldsymbol{H}_N & \vdots & \boldsymbol{H}_N\\ \cdots & \cdots & \cdots\\ \boldsymbol{H}_N & \vdots & -\boldsymbol{H}_N\end{bmatrix}=\boldsymbol{H}_2\otimes\boldsymbol{H}_N \tag{4-33}$$

式中，$\otimes$表示克罗内克（Kronecker）直积。2 阶哈达玛变换基等于：

$$\boldsymbol{H}_2=\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix} \tag{4-34}$$

采用式（4-33）生成哈达玛变换基的 MATLAB 代码如下：

```
function hadb = gethadb2(N)
n = log2(N);
if N - 2^floor(n) ~ = 0
    error('N is notintegeral power of 2');
end
hadb2 = [1 1;1  -1];
```

```
if n = =1
    hadb = hadb2;
else
    for i =2:n
        hadb2 = [hadb2 hadb2;hadb2  - hadb2];
    end
end
hadb = hadb2;
```

结合式（4-33）和式（4-34）易知哈达玛基是正交基；又由式（4-32）可知哈达玛基具有对称性。所以，反变换哈达玛基等于正变换哈达玛基。

4.1.4.2 数论变换

20 世纪 70 年代初，Rader 和 Agarwal 及 Burrus 等人提出了构造模 M 剩余类环 Z_M上的离散变换，即数论变换（**N**umber **T**heoretic **T**ransform），并把数论变换引入数字信号处理中。与余弦变换基相比，数论变换没有舍入误差，不需要存储三角函数，在相同变换长度下，速度优于余弦变换。而且数论变换本身就是整型变换，无需进行提升就可得到可逆的变换。

数论变换是在以正整数 M 为模的整数环 Z_M上定义的线形正交变换，所用运算法则是数论中同余运算，它在 Z_M上具有循环卷积特性，基本函数由整数的方幂构成。设 $x_i \in Z_M$，$i=0, 1, \cdots, N-1$，如果作用在序列 x_0，x_1，x_{N-1}上的一种变换为

$$X_k = \sum_{n=0}^{N-1} x_n \alpha^{nk} (\bmod M), \alpha \in Z_M, k = 0, 1, \cdots, N-1 \tag{4-35}$$

有如下形式的逆变换：

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k \alpha^{-nk} (\bmod M), \alpha \in Z_M, n = 0, 1, \cdots, N-1 \tag{4-36}$$

且具有循环卷积性质，则称式（4-35）为在 Z_M上长为 N 的数论变换，可表示成：

$$\boldsymbol{X} \equiv \boldsymbol{T} \cdot \boldsymbol{x} (\bmod M) \tag{4-37}$$

式中，$\boldsymbol{T}$ 是数论变换基，形式如下：

$$\boldsymbol{T} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & \alpha & \cdots & \alpha^{N-1} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \alpha^{N-1} & \cdots & \alpha^{(N-1)^2} \end{bmatrix} (\bmod M) \tag{4-38}$$

式（4-37）中，$\boldsymbol{x} = [x_0, x_1, \cdots, x_{N-1}]^{\mathrm{T}}$，$x_n \in Z_M (n=0, 1, \cdots, N-1)$ 且 $\alpha \in Z_M$。以上两式各元素选取的条件有：①M 是质数；②N 必须是 $M-1$ 的因子；③ α 是模 M 的 N 阶单位根，即 $\alpha^N = 1$（$\bmod M$），且 $\alpha^k \neq 1 (\bmod M)$（$k=1, 2, \cdots, N-1$）。

对比式（4-35）和式（4-36）可以看出，反数论变换基中的元素 $\beta = \alpha^{-1}$，又 $\alpha^{-1}(\bmod M) = \alpha^{N-1}(\bmod M)$，所以反数论变换基如下：

$$T^{-1} = \frac{1}{N}\begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & \alpha^{N-1} & \cdots & \alpha \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \alpha^{(N-1)^2} & \cdots & \alpha^{N-1} \end{bmatrix} (\bmod M) \tag{4-39}$$

实现构造数论变换 MATLAB 代码如下：

```
function tntb = gettnt(N,MM,a)
ijm = [0:N-1]'* [0:N-1];
tntb = mod(a.^ijm,MM);
tntb(tntb > MM/2) = tntb(tntb > MM/2) - MM;
```

4 阶数论变换（$M=5$，$\alpha=2$）基函数图形如图 4-5 所示。

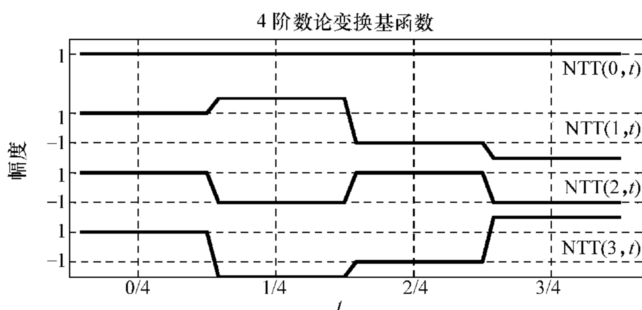


图 4-5　4 阶数论变换基函数图形

4.1.4.3　卡-洛变换

离散卡-洛变换是根据信号的统计特性性质进行正交展开。对于取样序列可看作随机场的一个样品，当数字信号用向量形式描述时，随机场中每个长度为 N 的信号都可表达成 N 维向量。如第 i 个信号序列可表示成向量 $\boldsymbol{x}_i = [x_i(0), x_i(1), \cdots, x_i(N-1)]^{\mathrm{T}}$，则由向量构成的随机场统计特性可由均值向量和自协方差矩阵表示如下：

$$\begin{aligned} \boldsymbol{m}_{\mathrm{x}} &= E\{\boldsymbol{x}\} \\ \boldsymbol{C}_{\mathrm{x}} &= E\{(\boldsymbol{x}-\boldsymbol{m}_{\mathrm{x}})(\boldsymbol{x}-\boldsymbol{m}_{\mathrm{x}})^{\mathrm{T}}\} = \boldsymbol{R}_{\mathrm{xx}} - \boldsymbol{m}_{\mathrm{x}}\boldsymbol{m}_{\mathrm{x}}^{\mathrm{T}} \end{aligned} \tag{4-40}$$

自协方差矩阵的特征向量 $\boldsymbol{a}_i$ 和特征值 λ_i 关系如下：

$$\boldsymbol{C}_{\mathrm{x}}\boldsymbol{a}_i = \lambda_i \boldsymbol{a}_i,\ i=0,\ 2,\ \cdots,\ N-1 \tag{4-41}$$

式中，$\boldsymbol{C}_{\mathrm{x}}$ 是 N 阶方阵；$\boldsymbol{a}_i$ 是 N 维向量。由于自协方差矩阵式是对称方阵，因此总可以找到 N 个相互正交的特征向量，从而构成了 N 维空间完备正交向量系。于是，对于 N 维空间中任意向量 $\boldsymbol{x}-\boldsymbol{m}_{\mathrm{x}}$ 可以正交展开成

$$\boldsymbol{x}-\boldsymbol{m}_{\mathrm{x}} = \sum_{i=0}^{M-1} g_i \boldsymbol{\alpha}_i,\ i=0,\ 1,\ 2\cdots,\ N-1 \tag{4-42}$$

式中，展开式系数 g_i 为

$$g_i = \boldsymbol{\alpha}_i^{\mathrm{T}}(\boldsymbol{x} - \boldsymbol{m}_{\mathrm{x}}) \tag{4-43}$$

按特征值递减顺序对下标 i 标号，即 $\lambda_0 \geqslant \lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_{N-1}$，则对应特征向量按下式组成正交变换矩阵 $\boldsymbol{A}$：

$$\boldsymbol{A} = [\boldsymbol{a}_0 \quad \boldsymbol{a}_1 \quad \cdots \quad \boldsymbol{a}_{N-1}]^{\mathrm{T}} \tag{4-44}$$

则式（4-41）和式（4-43）可分别表示成：

$$\begin{aligned} \boldsymbol{C}_{\mathrm{x}}\boldsymbol{A}^{\mathrm{T}} &= \boldsymbol{A}^{\mathrm{T}}\boldsymbol{\Lambda} \\ \boldsymbol{g} &= \boldsymbol{A}(\boldsymbol{x} - \boldsymbol{m}_{\mathrm{x}}) \end{aligned} \tag{4-45}$$

式中，$\boldsymbol{\Lambda}$ 是以 N 个特征值为元素构成的对主对角矩阵；$\boldsymbol{g}$ 是 N 个系数 g_i 构成的列向量即 $\boldsymbol{g} = [g_0, g_1, \cdots, g_{N-1}]^{\mathrm{T}}$。由式（4-44）表示的变换称为离散卡-洛变换，系数向量 $\boldsymbol{g}$ 是信号向量通过该正交变换得到的，称之为主成分信号向量。主成分信号向量 $\boldsymbol{g}$ 的均值 $\boldsymbol{m}_{\mathrm{g}} = E\{\boldsymbol{g}\} = \boldsymbol{A}E\{\boldsymbol{x}\} - \boldsymbol{A}\boldsymbol{m}_{\mathrm{x}} = 0$，所以自协方差矩阵：

$$\begin{aligned} \boldsymbol{C}_{\mathrm{g}} &= E\{(\boldsymbol{g} - \boldsymbol{m}_{\mathrm{g}})(\boldsymbol{g} - \boldsymbol{m}_{\mathrm{g}})^{\mathrm{T}}\} = E\{\boldsymbol{g}\boldsymbol{g}^{\mathrm{T}}\} \\ &= \boldsymbol{A}E\{(\boldsymbol{x} - \boldsymbol{m}_{\mathrm{x}})(\boldsymbol{x} - \boldsymbol{m}_{\mathrm{x}})^{\mathrm{T}}\}\boldsymbol{A}^{\mathrm{T}} \\ &= \boldsymbol{A}E\{\boldsymbol{x}\boldsymbol{x}^{\mathrm{T}}\}\boldsymbol{A}^{\mathrm{T}} = \boldsymbol{A}\boldsymbol{C}_{\mathrm{x}}\boldsymbol{A}^{\mathrm{T}} \\ &= \boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{\Lambda} \\ &= \boldsymbol{\Lambda} \end{aligned} \tag{4-46}$$

式（4-46）说明各主成分互不相关且每个特征值 λ_i 是 g_i 的方差，即

$$\lambda_i = E\{g_i^2\} = \sigma_{g_i}^2 \tag{4-47}$$

根据式（4-43），考虑到 $\boldsymbol{A}$ 是正交矩阵，故有

$$\boldsymbol{x} = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{g} + \boldsymbol{m}_{\mathrm{x}} \tag{4-48}$$

式（4-48）表明可用主成分来精确重建向量 $\boldsymbol{x}$。若用 M 个最大特征值所对应的特征向量组成变换矩阵 $\boldsymbol{A}_M$ 来代替 $\boldsymbol{A}$，即 $\boldsymbol{A}_M = [\boldsymbol{a}_0, \boldsymbol{a}_1, \cdots, \boldsymbol{a}_{M-1}]^{\mathrm{T}}$（$M < N$），则得系数向量为 $\boldsymbol{g}_M = \boldsymbol{A}_M(\boldsymbol{x} - \boldsymbol{m}_{\mathrm{x}})$，由此重构信号 $\boldsymbol{x}$ 得：

$$\hat{\boldsymbol{x}} = \boldsymbol{A}_M^{\mathrm{T}}\boldsymbol{g}_M + \boldsymbol{m}_{\mathrm{x}} \tag{4-49}$$

显然它不等于原来信号 $\boldsymbol{x}$ 值，误差为

$$\varepsilon = E\{\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|^2\} = E\left\{\left\|\sum_{i=M}^{N-1} g_i\boldsymbol{\alpha}_i\right\|^2\right\} = E\left\{\sum_{i=M}^{N-1} g_i^2\right\} = \sum_{i=M}^{N-1}\lambda_i^2 \tag{4-50}$$

因为是 λ_i 单调递减的，如果所选 M 个特征向量与 M 个最大特征值对应，那么用式（4-49）来重建信号向量的误差最小，所以在最小均方误差的意义上讲，离散卡-洛变换是最优的。但该变换矩阵 $\boldsymbol{A}$ 要用取样序列的自协方差矩阵来计算，因而随信号类型不同而各异，需求快速算法是值得研究的内容。离散卡-洛变换主要用于遥感多光谱信号中特征选择及数据压缩方面。

4.2　变换域全相位滤波器设计

4.2.1　DCT 域全相位滤波器

在诸多正交变换中，离散余弦变换（DCT）与 K-L 变换性能最接近，它们是在最小均方误差准则下进行的较理想变换。DCT 已被广泛地应用在图像压缩和特征提取等方面。国际标准建议 CCITT H.261、JPEG、MPEG 中都采用 8×8 的块做 DCT，但基于 DCT 的数字滤波尚不普遍。主要原因有：①做 DCT/IDCT 滤波处理时首先要对数据序列方块化，由此带来的问题是滤波值不但与方块的起点有关，而且在块与块的连接处留有方块化痕迹；②尽管 DCT 有快速算法. 但与通常数字滤波的卷积算法相比并不具有速度优势，而且在运算中还要进行许多数据的重排、移位、缓存等工作，硬件实现比较复杂；③在许多场合为避免波形失真需要滤波器具有线性相位特性，一般 DCT/IDCT 滤波不具有这种特性。为使 DCT 所具有的优良特性能在数字滤波中得到充分发挥，结合全相滤波的概念提出了 DCT 域 ap 系统设计。

基于 DCT 的 ap 系统是对经加窗处理后的输入信号作离散余弦变换，得到 DCT 域信号按传输特性 $\boldsymbol{H}$ 处理后经过 IDCT，并再次加窗处理后移位相加输出。系统原理同图 2-26 所示，对应的输出为

$$\begin{aligned} y(n) &= \sum_{i=0}^{N-1}[\boldsymbol{s}_{N-1-i}^{\mathrm{T}}\cdot \mathrm{diag}(\boldsymbol{B})\cdot \boldsymbol{C}^{\mathrm{T}}\cdot \boldsymbol{H}\cdot \boldsymbol{C}\cdot(\mathrm{diag}(\boldsymbol{F})\cdot \boldsymbol{x}_{n+i})] \\ &= \sum_{i=0}^{N-1}\left\{b(N-1-i)\sum_{j=0}^{N-1}\left[\left(\sum_{k=0}^{N-1}H_k c_{k,N-1-i}c_{k,j}\right)f(j)\right]\right\}x(n+i-N+1+j) \\ &= \sum_{i'=0}^{N-1}\sum_{j=0}^{N-1}[b(i')A(i',j)f(j)x(n-i'+j)] \qquad i'=N-1-i \\ &= \sum_{i=0}^{N-1}\sum_{k=-i}^{N-1-i}[b(i)A(i,k+i)f(k+i)x(n+k)] \qquad i=i',\ k=j-i' \\ &= T_0x(n)+\sum_{k=1}^{N-1}[T_kx(n+k)+T_{-k}x(n-k)] \end{aligned} \tag{4-51}$$

式中

$$A(i,j)=\sum_{m=0}^{N-1}\boldsymbol{H}_m c_{m,i}c_{m,j}=\boldsymbol{H}_0+\frac{1}{N}\sum_{m=1}^{N-1}\left[\cos\left(m\pi\cdot\frac{i+j+1}{N}\right)+\cos\left(m\pi\cdot\frac{i-j}{N}\right)\right]\cdot \boldsymbol{H}_m \tag{4-52}$$

不难证明，矩阵 $\boldsymbol{A}$ 满足：

$$A(i,j)=A(j,i)\quad A(N-1-i,N-1-j)=A(i,j) \tag{4-53}$$

$$\boldsymbol{T}_k = \sum_{i=0}^{N-1-k}[b(i)A(i,\ i+k)f(i+k)] \quad \boldsymbol{T}_{-k} = \sum_{i=0}^{N-1-k}[b(i+k)A(i+k,\ i)f(i)] \tag{4-54}$$

容易证明，DCT 域 ap 系统（简记为 apDCT）仍具有在 DFT 域时的诸多性质，如对 $\boldsymbol{H}$ 的线性性，线性相位的充分条件前后窗相等（即 $\boldsymbol{F}=\boldsymbol{B}$）。对于线性 apDCT，由上式可知，$\boldsymbol{T}_k=\boldsymbol{T}_{-k}(k=0,\ 1,\ \cdots,\ N-1)$，观察式（4-54），$\boldsymbol{T}_k$ 由卷积窗 $\boldsymbol{C}$ 和矩阵 $\boldsymbol{A}$ 按下列方式生成：

$$\begin{bmatrix} A_{00} & A_{01} & A_{01} & \cdots & A_{0N-2} & A_{0N-1} \\ A_{10} & A_{11} & A_{12} & \cdots & A_{1N-2} & A_{1N-1} \\ A_{20} & A_{21} & A_{22} & \cdots & A_{2N-2} & A_{2N-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ A_{N-20} & A_{N-21} & A_{N-22} & \cdots & A_{N-2N-2} & A_{N-2N-1} \\ A_{N-10} & A_{N-11} & A_{N-12} & \cdots & A_{N-1N-2} & A_{N-1N-1} \end{bmatrix} \rightarrow \begin{bmatrix} \boldsymbol{C}_{N-1} \\ \boldsymbol{C}_{N-2} \\ \boldsymbol{C}_{N-3} \\ \vdots \\ \boldsymbol{C}_2 \\ \boldsymbol{C}_1 \\ \boldsymbol{C}_0 \\ \vdots \end{bmatrix} \rightarrow \begin{bmatrix} \boldsymbol{T}_{N-1} \\ \boldsymbol{T}_{N-2} \\ \boldsymbol{T}_{N-3} \\ \vdots \\ \boldsymbol{T}_2 \\ \boldsymbol{T}_1 \\ \boldsymbol{T}_0 \\ \vdots \end{bmatrix}$$

$$\boldsymbol{k} \longrightarrow$$

$$[b_0 \quad b_1 \quad \cdots \quad b_{N-1}] * [b_0 \quad b_1 \quad \cdots \quad b_{N-1}] \quad =$$

矩阵 $\boldsymbol{A}$ 的计算根据式（4-53）的对称性可节省约 3/4 计算量，对于卷积窗 $\boldsymbol{C}$，如果基窗 $\boldsymbol{B}$ 具有中心对称性质，并令 $i'=N-1-k-i$ 则根据式（4-54）有

$$\begin{aligned} \boldsymbol{T}_k &= \sum_{i'=0}^{N-1-k} f(N-1-k-i')f(N-1-i')A(N-1-k-i',\ N-1-i') \\ &= \sum_{i'=0}^{N-1-k} f(i')f(k+i')A(i',\ k+i') \end{aligned} \tag{4-55}$$

上式表明，$\boldsymbol{T}_k$ 中 $N-k$ 个求和项满足中心对称，所以 $\boldsymbol{T}_k$ 又可以写为

$$\boldsymbol{T}_k = \begin{cases} 2\displaystyle\sum_{m=0}^{(N-k)/2-1} f(m)f(m+k)A(m,\ m+k) & k\text{ 是偶数} \\ t_0 + 2\displaystyle\sum_{m=0}^{(N-k-1)/2-1} f(m)f(m+k)A(m,\ m+k) & k\text{ 是奇数} \end{cases} \tag{4-56}$$

式中

$$t_0 = f\left(\frac{N-1-k}{2}\right)f\left(\frac{N-1+k}{2}\right)A\left(\frac{N-1-k}{2},\ \frac{N-1+k}{2}\right) \tag{4-57}$$

另一方面，apDCT 的传输特性由基序列 $\boldsymbol{H}$ 决定，确定过滤哪些余弦基信号分量。对于互补的 $\boldsymbol{H}_1$ 和 $\boldsymbol{H}_2$ 即 $\boldsymbol{H}_1+\boldsymbol{H}_2=\boldsymbol{1}$，有

$$\begin{aligned} &\sum_{m=0}^{N-1}(\boldsymbol{H}_1+\boldsymbol{H}_2)c_{m,\ i}c_{m,\ j} = \begin{cases} 1 & i=j \\ 0 & i\neq j \end{cases} \\ &\Rightarrow A_1(i,\ j) = \begin{cases} 1-A_2(i,\ j) & i=j \\ -A_2(i,\ j) & i\neq j \end{cases} \\ &\Rightarrow T_1(k) = \begin{cases} 1-T_2(k) & k=0 \\ -T_2(k) & k\neq 0 \end{cases} \end{aligned} \tag{4-58}$$

由（4-51）式可知，与 N 阶 apDCT 滤波器等效的 FIR 滤波器的单位冲击响应序列由 $\boldsymbol{T}_k$ 构成，即 $\boldsymbol{h}=[\boldsymbol{T}_{-N+1}\boldsymbol{T}_{-N+2}\cdots\boldsymbol{T}_{-1}\boldsymbol{T}_0\boldsymbol{T}_1\cdots\boldsymbol{T}_{N-2}\boldsymbol{T}_{N-1}]$，其传输特性为

$$H(j\omega) = \boldsymbol{T}_0 + 2\sum_{k=1}^{N-1}\boldsymbol{T}_k\cos(k\omega) \tag{4-59}$$

显然，apDCT 具有严格零相位。实现无窗、单窗和双窗 apDCT 与传统滤波器特性的 MATLAB 代码如下：

```
function T=apDCT(N)
ifnargin==0
   N=16;PB=8;
end
H=zeros(1,N);H(1:PB)=1;
wb=boxcar(N);wf=triang(N);wd=conv(wf,wf);
cosij=cosFun(N);A=cosij'* diag(H)* cosij;
for i=1:N
   t=repmat(diag(A,i-1),1,3);
   cwn=wb(i:N).* wb(1:N+1-i)/sum(wb);
   cws=wb(i:N).* wf(1:N+1-i)/sum(wf);
   cwd=wf(i:N).* wf(1:N+1-i)/wd(N);
   a=[cwn cws cwd];
   T(i,:)=sum(a.* t);
end
T=[T(1,:);2* T(2:end,:)];HH=freqFIR(N);
w=[0:2* pi/2^8:pi];k=0:N-1;kw=k'* w;ckw=cos(kw);
lw=length(w);
for i=1:3
   Ti=repmat(T(:,i),1,lw);HW(i,:)=sum(Ti.* ckw);
end
HW=cat(1,HW,abs(HH));
```

函数 cosFun 用于生成余弦基矩阵，freqFIR 是频率取样法响应值。运行结果如图 4-6 所示。

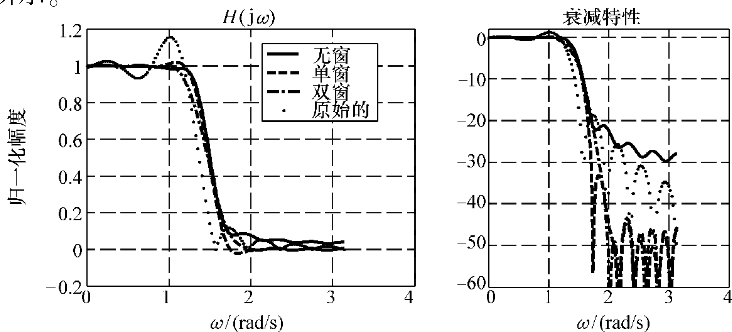


图 4-6　apDCT 与传统法特性曲线

由图 4-6 可以看出，apDCT 得到的滤波器特性无论在通带和阻带内波动均小于传统频率采样法，但过渡带宽度明显增加；对于不同基窗带来的性能改善及频率采样误差的深入剖析是值得研究的方向。

通过设置 $\boldsymbol{H}$ 可方便地设计子带滤波器，如以 Bartlett 为基窗的 2 倍频 16 阶双窗 apDCT 的 5 个子带特性如下：

$\boldsymbol{H}_0=[1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$；$\boldsymbol{H}_1=[0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$；

$\boldsymbol{H}_2=[0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$；$\boldsymbol{H}_3=[0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$；

$\boldsymbol{H}_4=[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1]$；

按照上述函数 apDCT 求得各子系统冲击相应如下：

$\boldsymbol{h}0$ = [0.0625 0.0612 0.0574 0.0519 0.0450 0.0375 0.0297 0.0223 0.0158 0.0106 0.0067 0.0039 0.0020 0.0009 0.0003 0.0000]

$\boldsymbol{h}1$ = [0.0246 0.0228 0.0178 0.0108 0.0032 −0.0038 −0.0089 −0.0117 −0.0121 −0.0107 −0.0083 −0.0057 −0.0033 −0.0016 −0.0005 −0.0001]

$\boldsymbol{h}2$ = [0.1303 0.1127 0.0682 0.0157 −0.0255 −0.0445 −0.0419 −0.0269 −0.0107 −0.0002 0.0038 0.0035 0.0019 0.0007 0.0001 0.0000]

$\boldsymbol{h}3$ = [0.2513 0.1138 −0.1147 −0.1638 −0.0452 0.0458 0.0359 0.0029 −0.0008 0.0047 0.0012 −0.0028 −0.0016 0.0002 0.0003 0.0000]

$\boldsymbol{h}4$ = [0.5313 −0.3105 −0.0287 0.0855 0.0225 −0.0350 −0.0148 0.0135 0.0078 −0.0044 −0.0033 0.0011 0.0010 −0.0002 −0.0001 0.0000]

各子带的传输特性如图 4-7 所示。

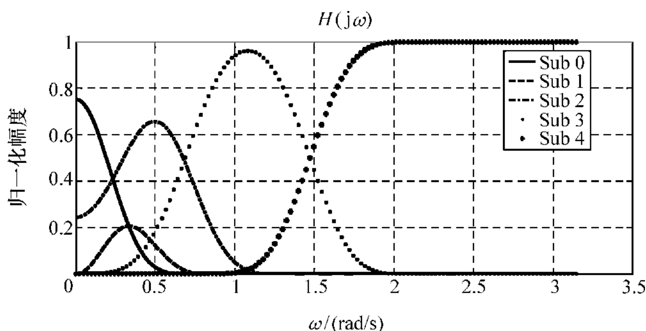


图 4-7　5 个子带特性曲线

实现上述结果的 MATLAB 代码如下：

```
function T = apDCTSB(N)
ifnargin = =0
    N =16;M = log2(N) +1;
end
```

```
HH = zeros(N,1);
H(:,1) = HH;H(1,1) = 1;
for i = 2:M
    H(:,i) = HH;H(2^(i - 2) + 1:2^(i - 1),i) = 1;
end
wf = triang(N);wd = conv(wf,wf);cosij = cosFun(N);
for i = 1:M
    A(:,:,i) = cosij'* diag(H(:,i))* cosij;
end
for k = 1:M
    for i = 1:N
        t = diag(A(:,:,k),i - 1);
        cwd = wf(i:N).* wf(1:N + 1 - i)/wd(N);
        T(i,k) = sum(cwd.* t);
    end
end
```

利用 apDCT 的各子带滤波器对图像的一行进行子带滤波的结果如图 4-8 所示。signal 为原始信号，**H**0 ~ **H**4 分别为三个带通子带信号，**H**0 为低通子带信号，**H**4 为高通信号。由图中可以看出，四个子带均无方块化痕迹。因此，apDCT 可以广泛应用于以离散余弦变换为基础的数字处理场合。在数字滤波、数字压缩以及信号的特征提取等方面可取代传统分块处理方法，利用其线性相位特性有效消除信号的波形失真。

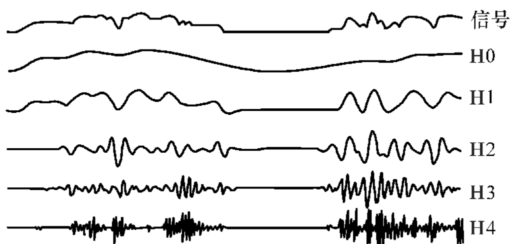


图 4-8　基于 apDCT 子带分解信号图

实现图 4-8 所示结果的 MATLAB 代码如下：

```
function T = apDCTSB(N)
ifnargin = = 0
    N = 16;M = log2(N) + 2;
end

HH = zeros(N,1);H(:,1) = ones(N,1);
H(:,2) = HH;H(1,2) = 1;
for i = 3:M
    H(:,i) = HH;
    H(2^(i - 2) + 1:2^(i - 1),i) = 1;
end
imgd = imread('test.tif');
```

```
sigin = double(imgd(:,size(imgd,2)/2,1));
for i =1:M
    apout(:,i) = apflt(H(:,i),sigin);
    apout(:,i) = apout(:,i)/max(abs(apout(:,i))) +(M - i) * 2;
end
plot(apout,'k','LineWidth',3);
set(gca,'XTickLabel',[],'YTickLabel',[],'Color',[1 1 1]);
axis off;
```

4.2.2 DWT 域全相位滤波器

近年来，由于数字技术、计算机技术以及 DSP 技术的迅速发展和广泛应用，迫使人们寻求更适用于这些新技术的分析工具，发现了还有很多其他函数在一定意义下比三角函数或指数函数更为优越，沃尔什（Walsh）函数就是其中之一。沃尔什函数系是一种完备正交函数系，每个函数只能取 +1 与 −1 两个值，就是所谓的二值正交函数。它的特点是图形呈矩形，有跳变特性，在跳变点间仅取上述两个数值，这与数字逻辑电路中的两个状态相对应，同时满足了完备和正交特性，从而可以进行级数展开与函数变换。因此，以沃尔什函数为基础所构成的变换，是实数加减运算，比傅里叶变换速度快。通常把沃尔什变换这类解析方法称为序域法，把基于这种变换法实现数字滤波的系统称为序率滤波器，又叫做列率滤波器。基于沃尔什变换的 ap（简记为 apDWT）是对经加窗处理后的输入信号作离散沃尔什变换，对得到的沃尔什域信号加权处理后再经过 IDWT，并再次加窗处理，移位相加输出。系统原理同图 2-26 所示，对应的输出为

$$
\begin{aligned}
y(n) &= \sum_{i=0}^{N-1}\left[\boldsymbol{s}_{N-1-i}^{\mathrm{T}}\cdot \mathrm{diag}(\boldsymbol{B})\cdot \boldsymbol{W}_{\mathrm{a}}^{\mathrm{T}}\cdot \boldsymbol{H}\cdot \boldsymbol{W}_{\mathrm{a}}\cdot(\mathrm{diag}(\boldsymbol{F})\cdot \boldsymbol{x}_{n+i})\right] \\
&= \frac{1}{\mathrm{N}}\sum_{i=0}^{N-1}\left\{b(i)\sum_{j=0}^{N-1}\left[\left(\sum_{k=0}^{N-1}\boldsymbol{H}_{\mathrm{k}}\mathrm{wal}(i,k)\mathrm{wal}(j,k)\right)f(j)\right]\right\}x(n-i+j) \\
&= \sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\left[b(i)A(i,j)f(j)x(n-i+j)\right] \\
&= \sum_{i=0}^{N-1}\sum_{k=-i}^{N-1-i}\left[b(i)A(i,k+i)f(k+i)x(n+k)\right] \\
&= \boldsymbol{T}_0x(n) + \sum_{k=1}^{N-1}\left[\boldsymbol{T}_kx(n+k) + \boldsymbol{T}_{-k}x(n-k)\right]
\end{aligned}
\tag{4-60}
$$

式中

$$
A(i,j) = \frac{1}{N}\sum_{k=0}^{N-1}\boldsymbol{H}_k\mathrm{wal}(i,k)\mathrm{wal}(j,k) = \frac{1}{N}\sum_{k=0}^{N-1}\boldsymbol{H}_k\mathrm{wal}(i\oplus j,k) \tag{4-61}
$$

$\boldsymbol{T}_k$的表达式与 apDCT 时相同。结合沃尔什变换的性质，易证明矩阵 $\boldsymbol{A}$ 仍具有式

（4-52）表示的主对称性和式（4-53）表示的次对称性。此外，apDWT 具有与 apDCT 相同的传输特性函数形式，因此也具有严格零相位的特性。沃尔什变换基具有正交基的一般性质，所以对于互补的系统特性 $\boldsymbol{H}_1$ 和 $\boldsymbol{H}_2$，对应的单位冲击响应序列 $\boldsymbol{T}_1$ 和 $\boldsymbol{T}_2$ 仍具有式（4-58）的结论。

参照函数 apDCT 容易写出 apDWT（不再列出），生成的特性曲线如图 4-9 所示。

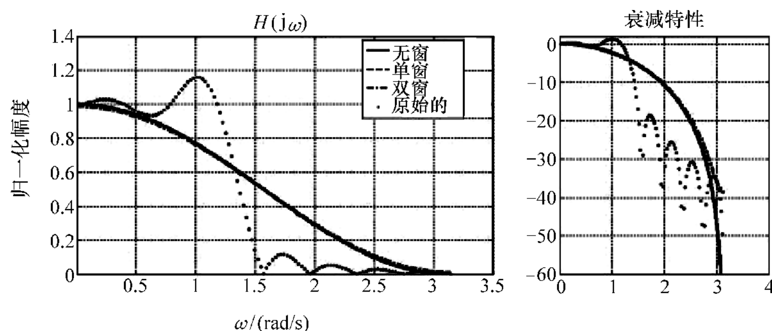


图 4-9　apDWT 与传统法特性曲线

与 apDCT 相同情形下，按照 2 倍频子带分解求得各 apDWT 子系统冲击相应如下：

***h*0** = [0. 0625 0. 0612 0. 0574 0. 0519 0. 0450 0. 0375 0. 0297 0. 0223 0. 0158 0. 0106 0. 0067 0. 0039 0. 0020 0. 0009 0. 0003 0. 0000]；

***h*1** = [0. 0625 0. 0405 0. 0216 0. 0060 －0. 0061 －0. 0145 －0. 0190 －0. 0195 －0. 0158 －0. 0106 －0. 0067 －0. 0039 －0. 0020 －0. 0009 －0. 0003 －0. 0000]；

***h*2** = [0. 1250 0. 0785 0. 0342 －0. 0057 －0. 0390 －0. 0230 －0. 0107 －0. 0028 0 0 0 0 0 0 0 0]；

***h*3** = [0. 2500 0. 0640 －0. 1132 －0. 0522 0 0 0 0 0 0 0 0 0 0 0 0]；

***h*4** = [0. 5000 －0. 1691 0 0 0 0 0 0 0 0 0 0 0 0 0 0]。

各子带的传输特性如图 4-10 所示。利用它对图像的一行进行子带滤波的结果如图 4-11 所示。

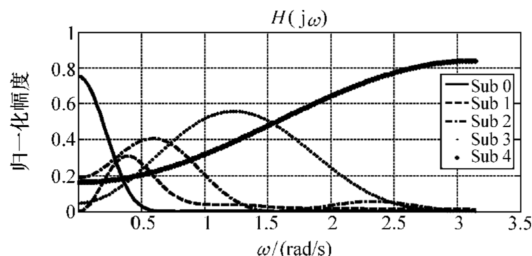


图 4-10　apDWT 各子带传输特性

signal 为原始信号，**H0** ~ **H4** 分别为三个带通子带信号，**H0** 为低通子带信号，**H4** 为高通信号。由图中可以看出，四个子带均无方块化痕迹。沃尔什变换具有某种能量集中性，原始数据分布越均匀，则变换后的数据越集中在矩阵边角上，因此在数据压缩方面比较实用。

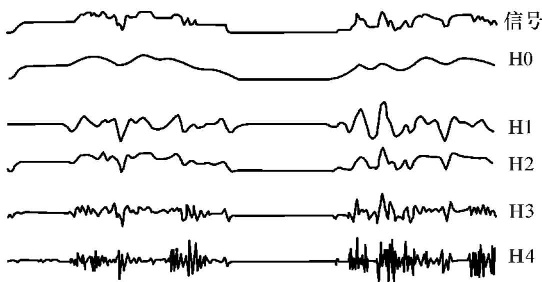


图 4-11　基于 apDWT 子带分解信号图

实际上，不论是何种正交变换，按照全相位信号处理方法得到的 ap 都可以对信号进行去除分段效应的滤波和处理，它是建立在新概念上的一种滤波系统。而由此带来对传统信号处理方法的改变则超出了它本身的定义，如自适应滤波、谱分析等都会得到良好的效果。

4.2.3　变换域全相位信号处理

对于傅里叶、余弦及沃尔什等正交变换的信号处理基本思路是结合输入信号特性和应用需求选择合适的正交基，在变换域内用传输特性 $\boldsymbol{H}$ 进行处理后再反变换得到对应的输出，其原理如图 4-12 所示。

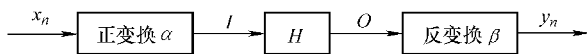


图 4-12　变换域信号处理图

对于 N 阶系统，时刻 n 输入 $\boldsymbol{x}_n=[x(n), x(n-1), \cdots, x(n-N+1)]^{\mathrm{T}}$，则根据图 4-12 所示的处理过程，输出 $\boldsymbol{y}_n$ 等于：

$$\boldsymbol{y}_n=\boldsymbol{\beta}\cdot[\boldsymbol{H}\otimes(\boldsymbol{\alpha}\cdot\boldsymbol{x}_n)] \tag{4-62}$$

式中，$\boldsymbol{\alpha}=[\alpha_{ij}]_{N\times N}$，$\boldsymbol{\beta}=[\beta_{ij}]_{N\times N}$；$\boldsymbol{H}=[H_0, H_1, \cdots, H_{N-1}]^{\mathrm{T}}$；$\otimes$表示对应元素相乘。将式（4-62）展开，对于 n 时刻输出中任意元素 $y(n-i)$ 等于：

$$\begin{aligned} y(n-i) &= \sum_{j=0}^{N-1}\sum_{k=0}^{N-1}\boldsymbol{\beta}_{i,j}\boldsymbol{H}_j\boldsymbol{\alpha}_{j,k}\cdot x(n-k) = \sum_{k=0}^{N-1}\left(\sum_{j=0}^{N-1}\boldsymbol{\beta}_{i,j}\boldsymbol{H}_j\boldsymbol{\alpha}_{j,k}\right)\cdot x(n-k) \\ &= \sum_{k=0}^{N-1}\left(\sum_{j=0}^{N-1}\boldsymbol{\alpha}_{j,i}^{*}\boldsymbol{H}_j\boldsymbol{\alpha}_{j,k}\right)\cdot x(n-k) \\ &= \sum_{k=0}^{N-1}\boldsymbol{T}_{i,k}\cdot x(n-k) \end{aligned} \tag{4-63}$$

矩阵 $\boldsymbol{T}$ 由正反正交基和特性向量 $\boldsymbol{H}$ 共同定义，具有共轭对称特性。如果以 $\boldsymbol{H}$ 的 N 个元素构成对角矩阵并也记为 $\boldsymbol{H}$，由图 4-12 容易得到：

$$\boldsymbol{T}=\boldsymbol{\beta}\cdot\boldsymbol{H}\cdot\boldsymbol{\alpha}=(\boldsymbol{\alpha}^{*})^{\mathrm{T}}\cdot\boldsymbol{H}\cdot\boldsymbol{\alpha} \tag{4-64}$$

全相位信号处理的基本思想是对 $x(n)$ 所在的 N 个分块作为输入时得到的 N 个输出的平均作为最终结果。可以用图 4-13 进行说明。

$$
\begin{array}{cccccccc}
x(n) & x(n-1) & x(n-2) & \cdots & x(n-N+2) & x(n-N+1) & \rightarrow & y_n(0) \\
x(n+1) & x(n) & x(n-1) & \cdots & x(n-N+3) & x(n-N+2) & \rightarrow & y_n(1) \\
x(n+2) & x(n+1) & x(n) & \cdots & x(n-N+4) & x(n-N+3) & \rightarrow & y_n(2) \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \rightarrow & \vdots \\
x(n+N-2) & x(n+N-3) & x(n+N-4) & \cdots & x(n) & x(n-1) & \rightarrow & y_n(N-2) \\
x(n+N-1) & x(n+N-2) & x(n+N-3) & \cdots & x(n+1) & x(n) & \rightarrow & y_n(N-1)
\end{array}
$$

图 4-13　全相位信号处理示意图

结合式（4-63），全相位处理后对应 $x(n)$ 的输出 $y(n)$ 为

$$
\begin{aligned}
y(n) &= \frac{1}{N}\sum_{m=0}^{N-1} y_n(m) = \frac{1}{N}\sum_{m=0}^{N-1}\sum_{i=0}^{N-1} \boldsymbol{T}_{m,i}\cdot x(n+m-i) \\
&= \frac{1}{N}\sum_{i=0}^{N-1}\sum_{j=-i}^{N-1-i} \boldsymbol{T}_{i+j,i}\cdot x(n+j),\quad j=m-i
\end{aligned}
\tag{4-65}
$$

按图 4-14 所示交换求和区间。

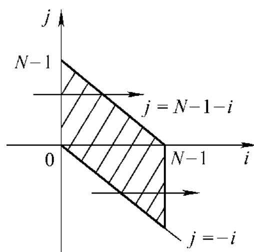


图 4-14　交换求和区间

式（4-65）可进一步表示为

$$
\begin{aligned}
y(n) &= \sum_{j=0}^{N-1} x(n+j)\sum_{i=0}^{N-1-j}\boldsymbol{T}_{i+j,i} + \sum_{j=-N+1}^{-1} x(n+j)\sum_{i=-j}^{N-1}\boldsymbol{T}_{i+j,i} \\
&= \sum_{j=0}^{N-1}\sum_{i=0}^{N-1-j}\frac{\boldsymbol{T}_{i+j,i}}{N}\cdot x(n+j) + \sum_{j=-N+1}^{-1}\sum_{i=0}^{N-1+j}\frac{\boldsymbol{T}_{i,i-j}}{N}\cdot x(n+j) \\
&= \boldsymbol{A}\cdot\boldsymbol{X}_1
\end{aligned}
\tag{4-66}
$$

式（4-66）就是全相位数字信号处理得到的输出与输入的数学关系表达。其中 $\boldsymbol{X}_1$ 是 1 维 N 阶全相位输入信号，即

$$
\begin{aligned}
\boldsymbol{X}_1 = [\,&x(n+N-1)\quad \cdots \quad x(n+2)\quad x(n+1)\quad x(n)\quad x(n-1) \\
&x(n-2)\quad \cdots \quad x(n-N+1)\,]^{\mathrm{T}}
\end{aligned}
\tag{4-67}
$$

矩阵 $\boldsymbol{A}$ 由矩阵 $\boldsymbol{T}$ 元素按下列方式构成：

$$
\boldsymbol{A}=\frac{1}{N}\begin{bmatrix}
\boldsymbol{T}_{N-1,0}\\
\boldsymbol{T}_{N-2,0}+\boldsymbol{T}_{N-1,1}\\
\vdots\\
\boldsymbol{T}_{2,0}+\boldsymbol{T}_{3,1}+\cdots+\boldsymbol{T}_{N-1,N-3}\\
\boldsymbol{T}_{1,0}+\mathrm{T}_{2,1}+\boldsymbol{T}_{3,2}+\cdots+\boldsymbol{T}_{N-1,N-2}\\
\boldsymbol{T}_{0,0}+\boldsymbol{T}_{1,1}+\boldsymbol{T}_{2,2}+\cdots+\boldsymbol{T}_{N-2,N-2}+\boldsymbol{T}_{N-1,N-1}\\
\boldsymbol{T}_{0,1}+\boldsymbol{T}_{1,2}+\boldsymbol{T}_{2,3}+\cdots+\boldsymbol{T}_{N-2,N-1}\\
\boldsymbol{T}_{0,2}+\boldsymbol{T}_{1,3}+\cdots+\boldsymbol{T}_{N-3,N-1}\\
\vdots\\
\boldsymbol{T}_{0,N-2}+\boldsymbol{T}_{1,N-1}\\
\boldsymbol{T}_{0,N-1}
\end{bmatrix}^{\mathrm{T}} \tag{4-68}
$$

把式（4-66）称为全相位变换，矩阵 $\boldsymbol{A}$ 称为全相位变换基，矩阵 $\boldsymbol{T}$ 则为全相位变换核。全相位变换基也可以通过核 $\boldsymbol{T}$ 顺时针旋转 45°后补零而构成，如图 4-15 所示。

$$
\boldsymbol{A}=\frac{1}{N}[1\ \ 1\ \ \cdots\ \ 1\ \ 1]\begin{bmatrix}
0 & \cdots & 0 & 0 & \boldsymbol{T}_{0,0} & 0 & 0 & \cdots & 0\\
0 & \cdots & 0 & \boldsymbol{T}_{1,0} & 0 & \boldsymbol{T}_{0,1} & 0 & \cdots & 0\\
\vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots\\
0 & \cdots & \boldsymbol{T}_{N/2,N/2-2} & 0 & \boldsymbol{T}_{N/2,N/2} & 0 & \boldsymbol{T}_{N/2-2,N/2} & \cdots & 0\\
\boldsymbol{T}_{N-1,0} & \cdots & 0 & \boldsymbol{T}_{N/2,N/2-1} & 0 & \boldsymbol{T}_{N/2-1,N/2} & 0 & \cdots & \boldsymbol{T}_{0,N-1}\\
0 & \cdots & \boldsymbol{T}_{N/2+1,N/2-1} & 0 & \boldsymbol{T}_{N/2+1,N/2-1} & 0 & \boldsymbol{T}_{N/2-1,N/2+1} & \cdots & 0\\
\vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots\\
0 & \cdots & 0 & \boldsymbol{T}_{N-1,N-2} & 0 & \boldsymbol{T}_{N-2,N-1} & 0 & \cdots & 0\\
0 & \cdots & 0 & 0 & \boldsymbol{T}_{N-1,N-1} & 0 & 0 & \cdots & 0
\end{bmatrix}
$$

图 4-15　旋转全相位核得到变换基矩阵 $\boldsymbol{A}$

全相位反变换可通过下面方式实现：

$$
\boldsymbol{X}_1=(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})^{-1}\cdot[\boldsymbol{A}^{\mathrm{T}}y(n)] \tag{4-69}
$$

综合以上，全相位变换的关键是构造基矩阵 $\boldsymbol{A}$，可参考以下步骤：

1）定义传输特性向量 $\boldsymbol{H}$；

2）按照式（4-64）求出全相位变换核矩阵 $\boldsymbol{T}$；

3）按照式（4-68）构造全相位变换基矩阵 $\boldsymbol{A}$；

4）按照式（4-66）进行全相位变换。

对于全通系统即 $\boldsymbol{H}=[1,\ 1,\ \cdots,\ 1]^{\mathrm{T}}$，无论哪种正交变换，对应的核 $\boldsymbol{T}$ 为单位矩阵，所以基矩阵 $\boldsymbol{A}$ 只有中间元素为 1，其余 $2N-2$ 个元素为 0，从而输出 $y(n)=x(n)$。对于一般情况，在傅里叶变换域时，有

$$
\begin{aligned}
\boldsymbol{T}_{i,\,k} &= \sum_{j=0}^{N-1} \boldsymbol{\alpha}_{j,\,i}^{*}\, \boldsymbol{H}_j\, \boldsymbol{\alpha}_{j,\,k},\ \alpha_{j,\,k} = \frac{1}{\sqrt{N}} W_N^{i\cdot k},\ W_N = \mathrm{e}^{-\mathrm{j}\frac{2\pi}{N}} \\
&= \frac{1}{N}\sum_{j=0}^{N-1} W_N^{k-i}\, \boldsymbol{H}_j \\
&= h(k-i)
\end{aligned} \tag{4-70}
$$

显然，核矩阵满主对称性：

$$
\boldsymbol{T}_{k,i} = \boldsymbol{T}_{N-1-i,\,N-1-k} \tag{4-71}
$$

容易求得全相位傅里叶变换基矩阵 $\boldsymbol{A}$ 等于：

$$
\begin{aligned}
\boldsymbol{A} = \frac{1}{N}[\,& h(1)\quad 2h(2)\quad \cdots\quad (N-1)h(N-1)\quad Nh(0) \\
& (N-1)h(1)\quad \cdots\quad 2h(N-2)\quad h(N-1)\,]
\end{aligned} \tag{4-72}
$$

线性 apDFT 系统即满足 $H(k) = H(N-k)$，由式（4-66）和式（4-72）求得传输特性如下：

$$
H(j\omega) = h(0) + \frac{2}{N}\sum_{k=1}^{N-1}(N-k)h(k)\cos(k\omega) \tag{4-73}
$$

对于 apDWT，全相位变换核等于：

$$
\begin{aligned}
\boldsymbol{T}_{i,\,k} &= \sum_{j=0}^{N-1} \boldsymbol{\alpha}_{j,\,i}^{*}\, \boldsymbol{H}_j\, \boldsymbol{\alpha}_{j,\,k},\ \alpha_{j,\,k} = \sqrt{\frac{1}{N}}\mathrm{Wal}(j,\,k) \\
&= \frac{1}{N}\sum_{j=0}^{N-1} \mathrm{Wal}(i \otimes k,\, j)\, \boldsymbol{H}_j \\
&= h(i \otimes k)
\end{aligned} \tag{4-74}
$$

由式（4-74）易知，apDWT 核同时具有主对称性（即 $\boldsymbol{T}_{i,k} = \boldsymbol{T}_{k,i}$）和次对称性（$\boldsymbol{T}_{N-1-i,\,N-1-k} = \boldsymbol{T}_{k,\,i}$）。相应的变换基具有中心对称性，4 阶 apDWT 变换基如下：

$$
\boldsymbol{A} = \frac{1}{4}[\,h(3)\quad 2h(2)\quad 2h(1)+h(3)\quad 4h(0)\quad 2h(1)+h(3)\quad 2h(2)\quad h(3)\,] \tag{4-75}
$$

对于 apDCT，全相位变换核无法合并，所以没有更具体的数学表达式，但基和核矩阵都具有上述对称性。

综合以上，变换域全相位数字信号处理一种可能实现如图 4-16 所示。

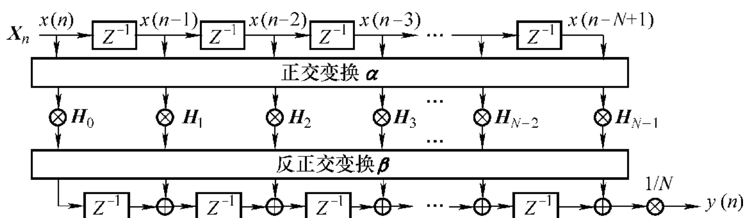


图 4-16　变换域全相位信号处理原理图

图 4-16 中 $\boldsymbol{X}_n$ 是 N 个与 $x(n)$ 相关的输入向量构成的矩阵，即

$$\boldsymbol{X}_n = \begin{bmatrix} \boldsymbol{x}_n^{\mathrm{T}} \\ \boldsymbol{x}_{n+1}^{\mathrm{T}} \\ \vdots \\ \boldsymbol{x}_{n+N-2}^{\mathrm{T}} \\ \boldsymbol{x}_{n+N-1}^{\mathrm{T}} \end{bmatrix} = \begin{bmatrix} x(n) & x(n+1) & \cdots & x(n+N-2) & x(n+N-1) \\ x(n-1) & x(n) & \cdots & x(n+N-3) & x(n+N-2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x(n-N+2) & x(n-N+3) & \cdots & x(n) & x(n+1) \\ x(n-N+1) & x(n-N+2) & \cdots & x(n-1) & x(n) \end{bmatrix} \tag{4-76}$$

构造传输特性矩阵 $\boldsymbol{H}$ 如下：

$$\boldsymbol{H} = \begin{bmatrix} H_0 & & & & \\ & H_1 & & 0 & \\ & & \ddots & & \\ & 0 & & H_{N-2} & \\ & & & & H_{N-1} \end{bmatrix}_{N\times N} \tag{4-77}$$

则全相位处理后输出等于：

$$\begin{aligned} \boldsymbol{Y} &= \boldsymbol{\beta} \cdot \boldsymbol{H} \cdot \boldsymbol{\alpha} \cdot \boldsymbol{X}_n^{\mathrm{T}} \\ y(n) &= \frac{1}{N}\sum_{i=0}^{N-1} \boldsymbol{Y}_{i,i} \end{aligned} \tag{4-78}$$

把式（4-78）进一步展开求得输出为

$$\begin{aligned} y(n) &= \frac{1}{N}\sum_{i=0}^{N-1}\left(\sum_{j=0}^{N-1}\sum_{k=0}^{N-1}\sum_{l=0}^{N-1}\boldsymbol{\beta}_{i,j}\boldsymbol{H}_{j,k}\boldsymbol{\alpha}_{k,l}(\boldsymbol{X}_n^{\mathrm{T}})_{l,i}\right), \text{only } k=j, \boldsymbol{H}_{j,k}\neq 0 \\ &= \frac{1}{N}\sum_{i=0}^{N-1}\left(\sum_{j=0}^{N-1}\sum_{l=0}^{N-1}\boldsymbol{\alpha}_{j,i}^{*}\boldsymbol{H}_{j,j}\boldsymbol{\alpha}_{j,l}x(n+i-l)\right), \boldsymbol{\beta}=(\boldsymbol{\alpha}^{*})^{\mathrm{T}} \text{and } (\boldsymbol{X}_n^{\mathrm{T}})_{l,i}=x(n+i-l) \\ &= \frac{1}{N}\sum_{i=0}^{N-1}\left(\sum_{l=0}^{N-1}\left(\sum_{j=0}^{N-1}\boldsymbol{\alpha}_{j,i}^{*}\boldsymbol{H}_{j,j}\boldsymbol{\alpha}_{j,l}\right)x(n+i-l)\right), \boldsymbol{T}_{i,l}=\sum_{j=0}^{N-1}\boldsymbol{\alpha}_{j,i}^{*}\boldsymbol{H}_{j,j}\boldsymbol{\alpha}_{j,l} \\ &= \frac{1}{N}\sum_{i=0}^{N-1}\left(\sum_{l=0}^{N-1}\boldsymbol{T}_{i,l}x(n+i-l)\right) \end{aligned} \tag{4-79}$$

这与式（4-65）结论相同，即全相位信号处理是对以 $x(n)$ 为中心的 $2N-1$ 个数据进行全相位变换。对于 apDCT、apDWT 和对称传输特性的 apDFT，变换基具有中心对称性，因此全相位系统具有严格零相位。与图 4-16 等效的全相位信号处理图如图 4-17 所示。

图中，$y(n)$ 是对应于 $x(n)$ 的输出，全相位处理把以 $x(n)$ 为中心的 $2N-1$ 个数据按照与 $x(n)$ 不同距离映射成不同权值 $\boldsymbol{A}_i$，而求得传输特性。对比模拟

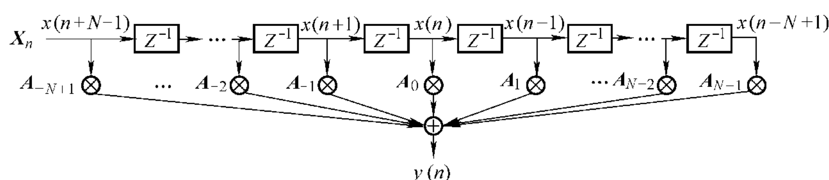


图 4-17　全相位信号处理原理图

系统的一一映射物理关系，可通过加窗以提高 $x(n)$ 在输入中的比重，一方面降低输出对 $x(n)$ 边缘数据的依赖，另一方面使全相位处理模拟化而改善传输特性。在图 4-16 的基础上，加入前后窗向量的变换域 ap 如图 4-18 所示。

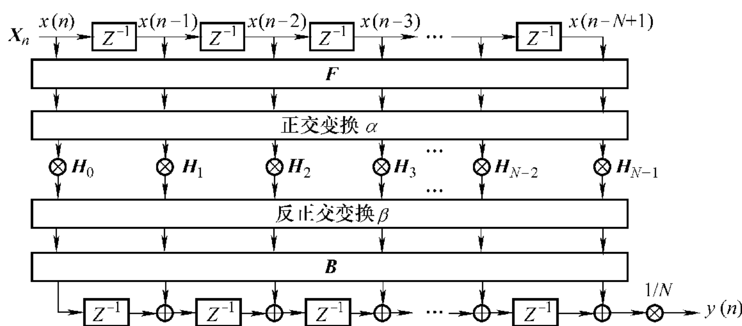


图 4-18　双窗全相位信号处理

按照与上面未加窗时推导方法可得到全相位变换核为

$$T_{i,l} = B_i F_l \sum_{j=0}^{N-1} \alpha_{j,i}^{*} H_{j,j} \alpha_{j,l} \tag{4-80}$$

对比式（4-68）可知双窗全相位变换基如下：

$$A = \frac{1}{N}\begin{bmatrix} B_{N-1}F_0 T_{N-1,0} \\ B_{N-2}F_0 T_{N-2,0} + B_{N-1}F_1 T_{N-1,1} \\ \vdots \\ B_2 F_0 T_{2,0} + B_3 F_1 T_{3,1} + \cdots + B_{N-1}F_{N-3}T_{N-1,N-3} \\ B_1 F_0 T_{1,0} + B_2 F_1 T_{2,1} + B_3 F_2 T_{3,2} + \cdots + B_{N-1}F_{N-2}T_{N-1,N-2} \\ F_0 B_0 T_{0,0} + F_1 B_1 T_{1,1} + F_2 B_2 T_{2,2} + \cdots + F_{N-2}B_{N-2}T_{N-2,N-2} + F_{N-1}B_{N-1}T_{N-1,N-1} \\ B_0 F_1 T_{0,1} + B_1 F_2 T_{1,2} + B_2 F_3 T_{2,3} + \cdots + B_{N-2}F_{N-1}T_{N-2,N-1} \\ B_0 F_2 T_{0,2} + B_1 F_3 T_{1,3} + \cdots + B_{N-3}F_{N-1}T_{N-3,N-1} \\ \vdots \\ B_0 F_{N-2}T_{0,N-2} + B_1 F_{N-1}T_{1,N-1} \\ B_0 F_{N-1}T_{0,N-1} \end{bmatrix}^{\mathrm{T}} \tag{4-81}$$

加窗全相位变换核都得到加权，$T_{i,j}$的权重等于$\boldsymbol{B}_i\boldsymbol{F}_j$，可以通过下列方式产生：

$\boldsymbol{B}$	$\boldsymbol{B}_0$	$\boldsymbol{B}_1$	$\cdots$	$\boldsymbol{B}_{N-1}$		
$\otimes$	$\boldsymbol{F}_0$	$\boldsymbol{F}_1$	$\cdots$	$\boldsymbol{F}_{N-1}$		
$\boldsymbol{F}$		$\boldsymbol{F}_0$	$\boldsymbol{F}_1$	$\cdots$	$\boldsymbol{F}_{N-1}$	→ 移动
$\Downarrow$	$\boldsymbol{B}_0\boldsymbol{F}_0$	$\boldsymbol{B}_1\boldsymbol{F}_1$	$\cdots$	$\boldsymbol{B}_{N-1}\boldsymbol{F}_{N-1}$		
$\boldsymbol{C}$	0	$\boldsymbol{B}_1\boldsymbol{F}_0$	$\boldsymbol{B}_2\boldsymbol{F}_1$	$\cdots$	0	

图 4-19　卷积窗生成过程

图 4-19 中$\otimes$表示对应元素相乘，处理过程类似卷积处理，因此把向量$\boldsymbol{C}$称为卷积窗。实际上，由上面推导 apDFT 可看出，全相位变换基权重就是前后窗卷积结果，对于无法合并核的 apDCT 或 apDWT 等变换域全相位信号处理，变换基权重是相互分离的。此外，由于引入卷积窗，对线性也产生了影响。下面分无窗、单窗和双窗三种情形进行讨论。

（1）无窗全相位处理。

除 apDFT 外其他正交变换基都是实矩阵，所以全相位核满足主对称性；apDFT 在传输特性满足对称条件下也使得全相位核具有主对称性。对无窗处理情形，$\boldsymbol{F}=\boldsymbol{B}=\mathbf{1}$，所以无窗 apDFT 具有严格零相位的充要条件是$H(N-k)=H(k)$（$k=0,1,\cdots,N-1$）；基于其他实正交基的全相位信号处理均具备严格零相位。

（2）单窗全相位处理。

假设$\boldsymbol{B}=\mathbf{1}$（$\boldsymbol{F}=\mathbf{1}$情形可类似推导），因为全相位核具有次对称性，观察式（4-81）易知全相位基具有中心对称的充要条件是$\boldsymbol{F}_k=\boldsymbol{F}_{N-1-k}$（$k=0,1,\cdots,N-1$）。

（3）双窗全相位处理。

全相位核具有主对称性和次对称性，由式（4-80）可知

$$\boldsymbol{T}_{l,i}=\boldsymbol{B}_l\boldsymbol{F}_i\sum_{j=0}^{N-1}\boldsymbol{\alpha}_{j,l}^{*}\boldsymbol{H}_{j,j}\boldsymbol{\alpha}_{j,i}=\boldsymbol{T}_{i,l}=\boldsymbol{B}_i\boldsymbol{F}_l\sum_{j=0}^{N-1}\boldsymbol{\alpha}_{j,i}^{*}\boldsymbol{H}_{j,j}\boldsymbol{\alpha}_{j,l} \tag{4-82}$$

对于 apDCT、apDWT 等实正交变换基，式（4-82）的求和项满足：

$$\sum_{j=0}^{N-1}\boldsymbol{\alpha}_{j,l}^{*}\boldsymbol{H}_{j,j}\boldsymbol{\alpha}_{j,i}=\sum_{j=0}^{N-1}\boldsymbol{\alpha}_{j,l}\boldsymbol{H}_{j,j}\boldsymbol{\alpha}_{j,i}=\sum_{j=0}^{N-1}\boldsymbol{\alpha}_{j,i}\boldsymbol{H}_{j,j}\boldsymbol{\alpha}_{j,l}=\sum_{j=0}^{N-1}\boldsymbol{\alpha}_{j,i}^{*}\boldsymbol{H}_{j,j}\boldsymbol{\alpha}_{j,l} \tag{4-83}$$

对于满足$H(N-k)=H(k)$的 apDFT，式（4-82）的求和项满足：

$$\sum_{j=0}^{N-1}\boldsymbol{\alpha}_{j,l}^{*}\boldsymbol{H}_{j,j}\boldsymbol{\alpha}_{j,i}=\sum_{j=0}^{N-1}\boldsymbol{\alpha}_{N-j,N-l}^{*}\boldsymbol{H}_{N-j,N-j}\boldsymbol{\alpha}_{N-j,N-i}=\sum_{j=0}^{N-1}\boldsymbol{\alpha}_{j,i}^{*}\boldsymbol{H}_{j,j}\boldsymbol{\alpha}_{j,l} \tag{4-84}$$

所以，式（4-82）成立的充要条件之一是前后窗相等即$\boldsymbol{F}=\boldsymbol{B}$。满足次对称性

时，式（4-83）和式（4-84）仍然成立，则式（4-82）成立的第二个重要条件是 $F(k)=F(\mathrm{N}-1-k)$ 且 $B(k)=B(N-1-k)$ 即前后窗满足中心对称性。

加窗的主要作用是提升信号主要成分在输出中的占比而进一步改善系统传输特性。上面介绍的加窗全相位信号处理方法中窗是固定不变的，如果始终把最大权值对准移动的信号分量 $x(n)$ 即形成滑动窗则可以再次改善特性。如何寻找窗以使特性在某种准则下最优是值得研究的内容。

4.3　小结

本章主要讨论了全相位信号处理在不同变换域中的特性和规律，首先介绍了常用的正交变换基如沃尔什和余弦基其基本性质；其次，分别推导沃尔什和余弦正交变换下全相位处理的数学规律并利用 MATLAB 编程实现了各自滤波特性对比和信号的子带分解；最后，通过两种思路对一维全相位数字信号处理的理论进行了基本整理，在定义了全相位变换、全相位核和基的基础上对 apDFT、apDCT 和 apDWT 的线性性展开讨论，并对前面 3 章的内容进行了拓展和深入总结。

第 5 章 二维全相位变换

模拟系统输出与输入之间建立的是一一对应的关系，传统数字信号处理则是与输入序列建立多对多的关系，而全相位方法是以序列作为输入而产生唯一输出。在一维信号处理时，输入是包含 $x(n)$ 的 N 个向量，可以将其视为以 $x(n)$ 为中心、长度为 $2N-1$ 的准 2 倍延拓，也可看做 N 个向量构成的矩阵。对于 $M \times N$ 二维信号的全相位处理，如果分别在 N 个长度为 M 的行信号上进行，虽然在行方向上可实现截断效应的改善，但列方向上由于忽略相关性而没有得到充分处理。二维的全相位信号处理应从纵横空间域上同时进行，即以分块重叠方式进行，如图 5-1 所示。

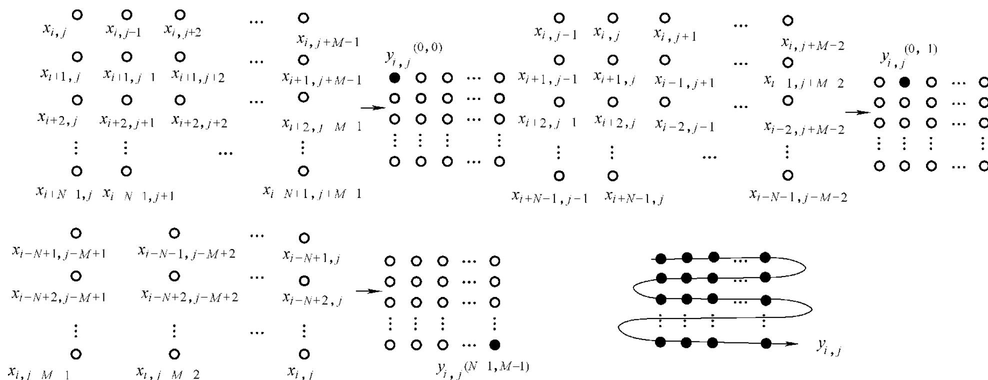


图 5-1　二维全相位处理示意图

由图 5-1 可以看出，元素 $x_{i,j}$ 在空间上 $N \times M$ 位置进行遍历，相应地得到 $N \times M$ 个子分量输出 $y_{i,j}^{(n,m)}$，最终结果等于所有子输出之和。与一维全相位不同的是，二维遍历的方式更加复杂，子系统传输特性序列 $\boldsymbol{H}$ 也是行列的二维函数，输出和输入均为位置的矩阵函数。

5.1　二维线性系统

图像表达的数学方式一般是二维矩阵，建立在矩阵表达上的图像处理同样可以在时域或变换域进行。同一维类似，基本系统分类如下。

1. 线性系统和非线性系统

具有叠加性和齐次性的系统称为线性系统。假如系统的特性可表示成对输入图像进行 $\boldsymbol{H}$ 运算，并令 $f_1(x,\ y)$ 与 $\boldsymbol{H}f_1(x,\ y)$ 以及 $f_2(x,\ y)$ 与 $\boldsymbol{H}f_2(x,\ y)$ 分别代表两对输入与输出图像。当系统满足

$$\boldsymbol{H}[f_1(x,\ y)+f_2(x,\ y)]=\boldsymbol{H}f_1(x,\ y)+\boldsymbol{H}f_2(x,\ y) \tag{5-1}$$

关系时，称系统具有叠加性。当系统满足

$$\boldsymbol{H}[kf(x,\ y)]=k\boldsymbol{H}f(x,\ y) \tag{5-2}$$

关系时，称系统具有齐次性。所以普遍来说，线性系统应满足以下关系：

$$\boldsymbol{H}\sum_{i=0}^{N-1}k_i f_i(x,\ y)=\sum_{i=0}^{N-1}k_i\boldsymbol{H}f_i(x,\ y) \tag{5-3}$$

凡是不满足叠加性或齐次性的系统属于非线性系统。

2. 位移不变系统与位移变化系统

位移不变系统的特性为：假设系统输入为 $f(x,\ y)$，所得输出 $\boldsymbol{H}f(x,\ y)$ 记为 $g(x,\ y)$；那么当输入为 $f(x-x_0,\ y-y_0)$ 时，对线性系统来说，其输出一般可表达成：

$$\boldsymbol{H}f(x-x_0,\ y-y_0)=g(x,\ x_0,\ y,\ y_0) \tag{5-4}$$

它与（x，y）及（x_0，y_0）均有关系。对于位移不变系统来说，由于系统参数本身不随位置改变，因此系统的输出与输入的位置没有关系，故位移不变系统的特性可表示成：若 $\boldsymbol{H}f(x,\ y)=g(x,\ y)$，故有

$$\boldsymbol{H}f(x-x_0,\ y-y_0)=g(x-x_0,\ y-y_0) \tag{5-5}$$

它表明当输入移动一个位置时，输出响应移动同样位置，但其形状不变。实际上，很多成像系统严格来说都是非线性的，但在一定条件下可看成线性的，且是位移不变的。本书中讨论的都是线性时不变系统模型。

从连续到数字转换的一个重要函数是冲激函数或 δ 函数，在二维情况下可写成：

$$\begin{cases}\displaystyle\iint_{-\infty}^{+\infty}\delta(x-x_0,\ y-y_0)\,\mathrm{d}x\mathrm{d}y=1\\ \delta(x-x_0,\ y-y_0)\big|_{\substack{x\neq x_0\\ y\neq y_0}}=0\\ \delta(x-x_0,\ y-y_0)\big|_{\substack{x\neq x_0\\ y\neq y_0}}=\infty\end{cases} \tag{5-6}$$

式（5-6）表示 δ 函数在出现（$x=x_0$，$y=y_0$）时为无限大，在其他各位置上其值为零，而它包含的体积是 1。可用图 5-2 所示数学模型从直观上进行理解此广义函数。

在图 5-2 中，取中心在 $(x_0,\ y_0)$ 点，面积为 ε^2 的方块，有

$$|x-x_0|\leqslant\varepsilon/2,\quad |y-y_0|\leqslant\varepsilon/2 \tag{5-7}$$

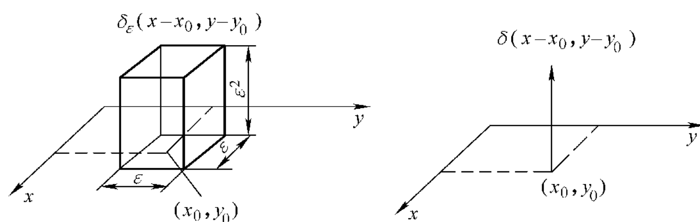


图 5-2　方柱体与 δ 函数

并考虑在此方块内的一个方柱体冲激：

$$\delta_\varepsilon(x-x_0,\ y-y_0)=\begin{cases}1/\varepsilon^2 & 方块内\\ 0 & 其他\end{cases} \tag{5-8}$$

它在方块面积内取常数值 $1/\varepsilon^2$，在其他取值为零。由式（5-7）和式（5-8）可见，对任何 ε 值，方柱体冲激 δ_ε（$x-x_0$，$y-y_0$）所包含的体积均为 1。当 ε 减少时，方柱体冲激的底面积变小，而其幅度增大。若使 ε 趋近于零，那么底面积 ε^2 也趋近于零，而幅度 $1/\varepsilon^2$ 必趋近于无限大，但方柱体冲激所包含的体积仍然为 1。这种极限的情况满足式（5-6）即 δ 函数，且其具有以下重要性质。

（1）筛选性质

考虑如下积分：

$$\begin{aligned}\lim_{\varepsilon\to0}\iint_{-\infty}^{+\infty}f(x,\ y)\delta_\varepsilon(x-x_0,\ y-y_0)\mathrm{d}x\mathrm{d}y &= \lim_{\varepsilon\to0}(1/\varepsilon^2)\int_{x_0-\varepsilon/2}^{x_0+\varepsilon/2}\int_{y_0-\varepsilon/2}^{y_0+\varepsilon/2}f(x,\ y)\mathrm{d}x\mathrm{d}y\\ &= f(x_0,\ y_0)\end{aligned} \tag{5-9}$$

式（5-9）对包含（x_0，y_0）点的任一积分区域都正确。

（2）δ 函数是偶函数

与上面的方法相同可得到：

$$\begin{aligned}\iint_{-\infty}^{+\infty}f(x,\ y)\delta(x_0-x,\ y_0-y)\mathrm{d}x\mathrm{d}y &= f(x_0,\ y_0)\Rightarrow\delta(x_0-x,\ y_0-y)\\ &= \delta(x-x_0,\ y-y_0)\end{aligned} \tag{5-10}$$

（3）卷积性质

根据式（5-9）和式（5-10）可得：

$$\iint_{-\infty}^{+\infty}f(\alpha,\ \beta)\delta(x-\alpha,\ y-\beta)\mathrm{d}\alpha\mathrm{d}\beta = f(x,\ y) \tag{5-11}$$

式（5-11）说明函数 $f(x,\ y)$ 与 $\delta(x,\ y)$ 的卷积再次产生 $f(x,\ y)$，记作 $f(x,\ y) * \delta(x,\ y) = f(x,\ y)$。并推广得到：

$$f(x,y) * \delta(x-\alpha,y-\beta) = f(x-\alpha,y-\beta) \tag{5-12}$$

$$f(x-x_1,y-y_1) * \delta(x-x_2,y-y_2) = f[x-(x_1+x_2),y-(y_1+y_2)] \tag{5-13}$$

（4）可分离性质

$$\delta(x-\alpha, y-\beta)=\delta(x-\alpha)\delta(y-\beta) \tag{5-14}$$

尽管 δ 函数并不是普通意义上定义的函数。但它有着很强的物理背景。如一幅图像是由无限多个点所组成，每一个像素均可看做是点光源。所以，任意图像 $f(x, y)$ 均可看做是无限多个点光源的组合。

根据卷积性质，若系统对输入图像就进行了线性运算 $\boldsymbol{H}$ 后将得到输出图像，结合线性系统叠加性和齐次性，可得如下表达式：

$$\begin{aligned} g(x, y) &= \boldsymbol{H}f(x, y) = \boldsymbol{H}\iint_{-\infty}^{+\infty} f(\alpha, \beta)\delta(x-\alpha, y-\beta)\mathrm{d}\alpha\mathrm{d}\beta \\ &= \iint_{-\infty}^{+\infty} \boldsymbol{H}\left[f(\alpha, \beta)\delta(x-\alpha, y-\beta)\right]\mathrm{d}\alpha\mathrm{d}\beta \\ &= \iint_{-\infty}^{+\infty} f(\alpha, \beta)\boldsymbol{H}\left[\delta(x-\alpha, y-\beta)\right]\mathrm{d}\alpha\mathrm{d}\beta \end{aligned} \tag{5-15}$$

令

$$\boldsymbol{H}\delta(x-\alpha, y-\beta)=h((x, \alpha; y, \beta) \tag{5-16}$$

式中，$h(x, \alpha; y, \beta)$ 称为系统单位冲激响应。将式（5-16）代入式（5-15）得

$$g(x,y) = \iint_{-\infty}^{+\infty} f(\alpha, \beta)h(x, \alpha; y, \beta)\mathrm{d}\alpha\mathrm{d}\beta \tag{5-17}$$

对于任意输入图像都可以按照式（5-17）计算得到输出图像即单位冲激响应完全表征了线性系统的特性。若系统是位移不变的，则系统输出与输入的位置无关，即

$$\boldsymbol{H}\delta(x-\alpha, y-\beta)=h(x-\alpha, y-\beta) \tag{5-18}$$

于是式（5-17）可简化成卷积形式：

$$g(x, y) = \iint_{-\infty}^{+\infty} f(\alpha, \beta)h(x-\alpha, y-\beta)\mathrm{d}\alpha\mathrm{d}\beta \tag{5-19}$$

在图像处理中，通常是把成像系统描述成线性位移不变的，因为这种情况基本符合客观实际。系统能描述成线性位移不变具有很多优点，它可采用业已成熟的线性系统理论的许多手段进行分析，易于问题的数学处理。

式（5-19）给出了连续图像函数的卷积形式，用计算机实现时，仍然需要数字化后才能进行运算。假设对图像 $f(x, y)$ 和系统冲激响应 $h(x, y)$ 均匀取样，样本数分别为 $A\times B$ 和 $C\times D$，并在卷积前都周期性延拓成 $M\times N$ 样本，即

$$f_e(x,y)=\begin{cases}f(x,y) & 0\leqslant x\leqslant A-1,\ 0\leqslant y\leqslant B-1\\ 0 & A\leqslant x\leqslant M-1,\ B\leqslant y\leqslant N-1\end{cases}$$
$$h_e(x,y)=\begin{cases}h(x,y) & 0\leqslant x\leqslant C-1,\ 0\leqslant y\leqslant D-1\\ 0 & C\leqslant x\leqslant M-1,\ D\leqslant y\leqslant N-1\end{cases} \tag{5-20}$$

式中，$f_e(x,y)$ 和 $h_e(x,y)$ 都是二维周期性离散函数，在行列方向上的周期长度分别为 M、N。这两个函数卷积等于：

$$g_e(x,y)=\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}f_e(m,n)h_e(x-m,y-n)\quad 0\leqslant x\leqslant M-1;\ 0\leqslant y\leqslant N-1 \tag{5-21}$$

显然，$g_e(x,y)$ 也是周期性离散函数，它在行列方向上的周期长度同样分别等于 M、N。为使卷积不产生卷绕效应，必须选择 $M\geqslant A+C-1$ 和 $N\geqslant B+D-1$。

若含有 $M\times N$ 点阵的 $f_e(x,y)$ 和 $h_e(x,y)$ 用列叠列的方法表示成 MN 维列向量，则式（5-21）可表示为

$$\boldsymbol{g}=\boldsymbol{H}\cdot\boldsymbol{f} \tag{5-22}$$

式中，$\boldsymbol{g}$ 及 $\boldsymbol{f}$ 是 $MN\times 1$ 维列向量；$\boldsymbol{H}$ 是 $MN\times MN$ 维方阵，包含 M^2 个大小为 $N\times N$ 的分区，按分区排列的 $\boldsymbol{H}$ 方阵可写为

$$\boldsymbol{H}=\begin{bmatrix}\boldsymbol{H}_0 & \boldsymbol{H}_{M-1} & \boldsymbol{H}_{M-2} & \cdots & \boldsymbol{H}_1\\ \boldsymbol{H}_1 & \boldsymbol{H}_0 & \boldsymbol{H}_{M-1} & \cdots & \boldsymbol{H}_2\\ \boldsymbol{H}_2 & \boldsymbol{H}_1 & \boldsymbol{H}_0 & \cdots & \boldsymbol{H}_3\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ \boldsymbol{H}_{M-1} & \boldsymbol{H}_{M-2} & \boldsymbol{H}_{M-3} & \cdots & \boldsymbol{H}_0\end{bmatrix}_{MN\times MN} \tag{5-23}$$

每一个分区 $\boldsymbol{H}_j$ 的元素是由 $h(x,y)$ 的第 j 行扩展函数组成，即

$$\boldsymbol{H}_j=\begin{bmatrix}h_e(j,0) & h_e(j,N-1) & h_e(j,N-2) & \cdots & h_e(j,1)\\ h_e(j,1) & h_e(j,0) & h_e(j,N-1) & \cdots & h_e(j,2)\\ h_e(j,2) & h_e(j,1) & h_e(j,0) & \cdots & h_e(j,3)\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ h_e(j,N-1) & h_e(j,N-2) & h_e(j,N-3) & \cdots & h_e(j,0)\end{bmatrix}_{N\times N} \tag{5-24}$$

5.2 二维变换正交基

设数字图像 $f(x,y)$ 由 $N\times N$ 个像素组成，若用 $h(x,y;i,j)$ 表示变换核，则图像由空间域 (x,y) 平面经过正交变换到变换域 (i,j) 可表示成

$$T(i,j)=\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}f(x,y)h(x,y;i,j) \tag{5-25}$$

可以把$f(x, y)$和$T(i, j)$用列叠列方式表示长度为N^2的列向量$\boldsymbol{f}$和$\boldsymbol{T}$，把$h(x, y; i, j)$排列成N^2方阵$\boldsymbol{H}$，仿照式（5-22）式（5-25）可表示为

$$\boldsymbol{T} = \boldsymbol{H} \cdot \boldsymbol{f} \tag{5-26}$$

如果变换核可按照坐标自变量分离成两个核，即

$$h(x, y; i, j) = u(x, i)v(y, j) \tag{5-27}$$

则式（5-25）可写成：

$$T(i, j) = \sum_{x=0}^{N-1} \left[\sum_{y=0}^{N-1} f(x, y)v(y, j) \right] u(x, i) \tag{5-28}$$

一般二维信号$f(x, y)$取样值可用$N \times N$个元素组成的矩阵$\boldsymbol{X}$表示成：

$$\boldsymbol{X} = \begin{bmatrix} X_{0,0} & X_{0,1} & \cdots & X_{0,N-1} \\ X_{1,0} & X_{1,1} & \cdots & X_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ X_{N-1,0} & X_{N-1,1} & \cdots & X_{N-1,N-1} \end{bmatrix}_{N \times N} \tag{5-29}$$

则图像矩阵$\boldsymbol{X}$二维线性变换$\boldsymbol{T}$可写成：

$$\boldsymbol{T} = \frac{1}{N^2}\boldsymbol{U}^{\mathrm{T}}\boldsymbol{X}\boldsymbol{V},\ \boldsymbol{U}^{\mathrm{T}}\boldsymbol{U} = N\boldsymbol{I} \text{ 和 } \boldsymbol{V}\boldsymbol{V}^{\mathrm{T}} = N\boldsymbol{I} \tag{5-30}$$

变换$\boldsymbol{T}$又称为图像的“酉变换域”，$\boldsymbol{U}$和$\boldsymbol{V}$称为酉算子，$\boldsymbol{I}$是单位矩阵。由式（5-30）易知逆变换为

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{T}\boldsymbol{V}^{\mathrm{T}} \tag{5-31}$$

若$\boldsymbol{U}$和$\boldsymbol{V}$可写成如下形式：

$$\begin{aligned} \boldsymbol{U} &= [\boldsymbol{u}_0 \quad \boldsymbol{u}_1 \quad \cdots \quad \boldsymbol{u}_{N-1}] \\ \boldsymbol{V} &= [\boldsymbol{v}_0 \quad \boldsymbol{v}_1 \quad \cdots \quad \boldsymbol{v}_{N-1}] \end{aligned} \tag{5-32}$$

式（5-32）中$\boldsymbol{u}_i$和$\boldsymbol{v}_i$是由酉算子各列组成的列向量，这样式（5-31）就可写成：

$$\boldsymbol{X} = [\boldsymbol{u}_0 \quad \boldsymbol{u}_1 \quad \cdots \quad \boldsymbol{u}_{N-1}]\boldsymbol{T} \begin{bmatrix} \boldsymbol{v}_0^{\mathrm{T}} \\ \boldsymbol{v}_1^{\mathrm{T}} \\ \vdots \\ \boldsymbol{v}_{N-1}^{\mathrm{T}} \end{bmatrix} \tag{5-33}$$

若酉变换域矩阵$\boldsymbol{T}$写成如下求和形式：

$$\boldsymbol{T} = \begin{pmatrix} t_{00} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix}_{N \times N} + \begin{pmatrix} \boldsymbol{0} & t_{0,N-1} \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix}_{N \times N} + \cdots + \begin{pmatrix} \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & t_{N-1,N-1} \end{pmatrix}_{N \times N} \tag{5-34}$$

则式（5-33）可写成下列外积形式：

$$\boldsymbol{X} = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} t_{i,j}\boldsymbol{u}_i \boldsymbol{v}_j^{\mathrm{T}} \tag{5-35}$$

$M \times 1$向量$\boldsymbol{U}$和$N \times 1$向量$\boldsymbol{V}$的外积$\boldsymbol{U}\boldsymbol{V}^{\mathrm{T}}$为$M \times N$矩阵，元素$\boldsymbol{u}_i\boldsymbol{v}_j^{\mathrm{T}}$代表变换的基

本图像。式（5-35）说明任意图像可表示成以变换域系数 $t_{i,j}$ 为加权的基本图像组合。

5.2.1 二维傅里叶变换

傅里叶变换是变换域分析的一种，在图像处理中是一种有效且重要的分析手段，因为有些处理方法直接和滤波概念相联系，需借助傅里叶变换把空间域信号映射到空间频率域上来分析。此外，借助傅里叶变换，可简化计算或作某种特殊应用如数据压缩等。离散傅里叶变换对建立了空间位置函数 $f(x, y)$ 与空间频率函数 $F(u, v)$ 之间的转换关系，正反变换如下：

$$T(u, v) = \frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x, y)\exp\left[-\mathrm{j}2\pi(ux+vy)/N\right]$$
$$u, v = 0, 1, 2, \cdots, N-1$$
$$f(x, y) = \frac{1}{N}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} T(u, v)\exp\left[\mathrm{j}2\pi(ux+vy)/N\right] \tag{5-36}$$
$$x, y = 0, 1, 2, \cdots, N-1$$

这里假设图像由 $N \times N$ 个元素组成。由式（5-36）容易推导出傅里叶变换基矩阵如下：

$$F_{\mathrm{N}}(u,x) = \left[\exp(-\mathrm{j}2\pi ux/N)\right]_{N\times N}$$
$$= \exp(-\mathrm{j}2\pi/N)\begin{pmatrix} 0 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & N-1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & N-1 & 2(N-1) & (N-1)^2 \end{pmatrix} \tag{5-37}$$

用矩阵表示式（5-36）如下：

$$\boldsymbol{T}_{\mathrm{F}} = \boldsymbol{F}_{\mathrm{N}}\boldsymbol{X}\boldsymbol{F}_{\mathrm{N}} \tag{5-38}$$

式（5-38）对应的逆变换为

$$\boldsymbol{X} = \boldsymbol{F}_{\mathrm{N}}^{*}\ \boldsymbol{T}_{\mathrm{F}}\ \boldsymbol{F}_{\mathrm{N}}^{*} \tag{5-39}$$

式中，$*$ 表示共轭。以 $N=4$ 为例，为讨论方便，元素坐标原点设在矩阵 $\boldsymbol{X}$ 和 $\boldsymbol{T}$ 的中心。逆变换基矩阵等于：

$$\boldsymbol{F}_{\mathrm{N}}^{*} = \frac{\exp(-\mathrm{j}2\pi/4)}{4}\begin{pmatrix} 0 & 2 & 0 & 2 \\ 2 & 1 & 0 & 3 \\ 0 & 0 & 0 & 0 \\ 2 & 3 & 0 & 1 \end{pmatrix} \tag{5-40}$$

由式（5-39）可得到傅里叶基本图像见表 5-1。

表 5-1　4 阶傅里叶基本图像指数

	0	2	0	2		2	3	0	1		0	0	0	0		2	1	0	3
0	0	2	0	2		2	3	0	1		0	0	0	0		2	1	0	3
2	2	0	2	0		0	1	2	3		2	2	2	2		0	3	2	1
0	0	2	0	2		2	3	0	1		0	0	0	0		2	1	0	3
2	2	0	2	0		0	1	2	3		2	2	2	2		0	3	2	1
	F (−2, −2)					***F*** (−1, −2)					***F*** (0, −2)					***F*** (1, −2)			
2	2	0	2	0		0	1	2	3		2	2	2	2		0	3	2	1
3	3	1	3	1		1	2	3	0		3	3	3	3		1	0	3	2
0	0	2	0	2		2	3	0	1		0	0	0	0		2	1	0	3
1	1	3	1	3		3	0	1	2		1	1	1	1		3	2	1	0
	F (−2, −1)					***F*** (−1, −1)					***F*** (0, −1)					***F*** (1, −1)			
0	0	2	0	2		2	3	0	1		0	0	0	0		2	1	0	3
0	0	2	0	2		2	3	0	1		0	0	0	0		2	1	0	3
0	0	2	0	2		2	3	0	1		0	0	0	0		2	1	0	3
0	0	2	0	2		2	3	0	1		0	0	0	0		2	1	0	3
	F (−2, 0)					***F*** (−1, 0)					***F*** (0, 0)					***F*** (1, 0)			
2	2	0	2	0		0	1	2	3		2	2	2	2		0	3	2	1
1	1	3	1	3		3	0	1	2		1	1	1	1		3	2	1	0
0	0	2	0	2		2	3	0	1		0	0	0	0		2	1	0	3
3	3	1	3	1		1	2	3	0		3	3	3	3		1	0	3	2
	F (−2, 1)					***F*** (−1, 1)					***F*** (0, 1)					***F*** (1, 1)			

记 $W_N = \exp(-j2\pi/N)$，则傅立叶基本图像由 $W_N\boldsymbol{F}(i, j)$ 得到，共有 16 个傅里叶基本图像，每个图像是 4×4 个像素矩阵。需要指出的是，傅里叶基本图像都是复图像。MATLAB 实现的基本图像和傅里叶谱图的代码如下：

```
function out = dft2Fun(N)
if nargin = =0
  N = 4;low = (N - mod(N,2))/2;high = low - 1 + mod(N,2);M = 64/N;
end
ex = [ - low:high]'* [ - low:high],fbase = mod(ex + N,N);
fconbase = mod(N - fbase,N);out = exp( - j* 2* pi/N* fconbase);
for v = 1:N
    for u = 1:N
        n = (v - 1)* N + u;r = repmat(fconbase(:,v),1,N);
        c = repmat(fconbase(u,:),N,1);fbimgs(:,:,n) = mod(r + c,N);
        Tb(:,:,n) = (exp( - j* 2* pi/N* fbase)* ...
            exp( - j* 2* pi/N* fbimgs(:,:,n))* ...
            exp( - j* 2* pi/N* fbase))/N^2;
```

```
    end
end

if nargin = =0
    for v =1:N
        for u =1:N
            n = (v -1)* N +u;fb =exp( -j* 2* pi/N* fbimgs(:,:,n));
            fbr =real(fb) +1;fbi =imag(fb) +1;tb =abs(Tb(:,:,n));
            for nr =1:N
                for nc =1:N
                    r = (nr -1)* M +1;c = (nc -1)* M +1;
                    fbrimg(r:r +M -1,c:c +M -1,n) =ones(M,M)* fbr(nr,nc);
                    fbiimg(r:r +M -1,c:c +M -1,n) =ones(M,M)* fbi(nr,nc);
                    tbimg(r:r +M -1,c:c +M -1,n) =ones(M,M)* tb(nr,nc);
                end
            end
            fbrimg(:,:,n) =fbrimg(:,:,n)/2;fbiimg(:,:,n) =fbiimg(:,:,n)/2;
        end
    end
    imgs(:,:,:,1) =fbrimg;imgs(:,:,:,2) =fbiimg;imgs(:,:,:,3) =tbimg;
    for in =1:3
        figure(in);
        for v =1:N
            for u =1:N
                n = (v -1)* N +u;
                subplot(N,N,n),imshow(imgs(:,:,n,in));
            end
        end
    end
end
```

运行结果如图 5-3 与图 5-4 所示。

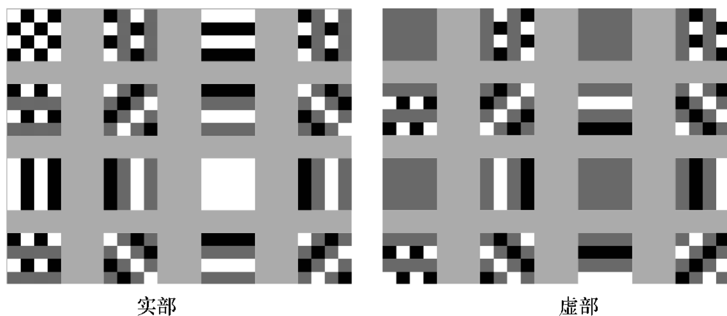


图 5-3　4 阶傅里叶基本图像

由图 5-3 可以看出，实部和虚部相位差是 $\pi/2$，对于 4 阶傅里叶变换的空间单位频率是 $2\pi/4$，即实部和虚部按隔行或隔列对准。由图 5-4 可以看出，基本图像的频谱是分布在 (u, v) 处的单点频率，分别表示在行列方向上的空间频率。

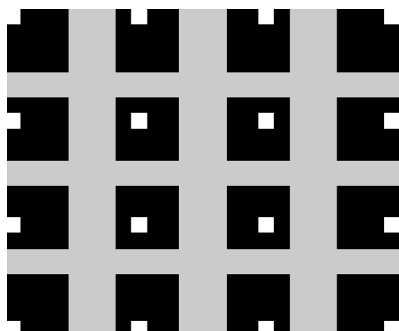


图 5-4　4 阶傅里叶基本图像频谱图

离散傅里叶变换建立了空间位置函数 $f(x, y)$ 与空间频率函数 $T(u, v)$ 之间的转换关系。数字图像处理中经常要用到图像的空间域与空间频域之间的对应关系及其相互转换规律。

（1）线性

令 $\mathscr{T}\{f(x, y)\}$ 表示离散函数 $f(x, y)$ 的傅里叶变换，即 $T(u, v)=\mathscr{T}\{f(x, y)\}$，则式（5-36）定义的傅里叶变换为

$$\begin{aligned}\mathscr{T}\{af_1(x, y)+bf_2(x, y)\} &= a\mathscr{T}\{af_1(x, y)\}+b\mathscr{T}\{f_2(x, y)\}\\ &= aT_1(u, v)+bT_2(u, v)\end{aligned} \tag{5-41}$$

（2）180°旋转

$$\mathscr{T}\{\mathscr{T}\{f(x, y)\}\}=f(-x, -y) \tag{5-42}$$

（3）改变符号

$$\mathscr{T}\{f(-x, -y)\}=T(-u, -v) \tag{5-43}$$

（4）改变标度因子

$$\mathscr{T}\{f(ax, by)\}=T(u/a, v/b) \tag{5-44}$$

（5）空间位置平移

如果 $f(x, y)$ 在 x 轴、y 轴上分别平移了 x_0、y_0，则有

$$\mathscr{T}\{f(x-x_0, y-y_0)\}=T(u, v)\exp[-j2\pi(ux_0/M+vy_0/N)] \tag{5-45}$$

（6）空间频率平移

$$\mathscr{T}\{f(x, y)\exp[j2\pi(u_0x/M+v_0y/N)]\}=T(u-u_0, v-v_0) \tag{5-46}$$

（7）循环卷积

如果 $f_P(x, y)$ 和 $g_P(x, y)$ 是周期为 (N, N) 的二维函数，做循环卷积 $f_P(x, y) \odot g_P(x, y)$，则有

$$\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} f_P(m, n)g_P(x-m, y-n) \rightarrow F(u, v)G(u, v) \tag{5-47}$$

对于坐标尺寸为 $M\times N$ 两个非周期函数 $f(x, y)$ 和 $g(x, y)$，则需要先进行周期延拓成 $N\times N$ 周期函数。若 $N\geqslant 2M-1$，则先补零成为坐标尺寸为 $N\times N$ 的函数，再做周期延拓成两个周期函数 $f_P(x, y)$ 和 $g_P(x, y)$，若 K 和 L 为整数，

则有

$$f_P(KN+m,\ LN+n)=f_P(m,\ n)\quad g_P(KN+m,\ LN+n)=g_P(m,\ n)\qquad(5\text{-}48)$$

显然在一个周期内的循环卷积等于普通卷积。验证上述性质的 MATLAB 代码如下：

```
function out = dftXZ(N)
if nargin = = 0
    N = 5; low = (N - mod(N,2))/2; high = low - 1 + mod(N,2);
    preci = exp( - 10);
    chstr = { 'Linearity','rotation 180 degree','plus to minus',...
      'posi move','freq move','cyclic conv' };
end
len = length(chstr); res = zeros(N,N,2* len);
ex = [ - low:high]'* [ - low:high]; fbase = mod(ex + N,N);
fconbase = mod(N - fbase,N); out = exp( - j* 2* pi/N* fbase);
[res(:,:,1),res(:,:,2)] = linearXZ(N,out);
[res(:,:,3),res(:,:,4)] = rotate180(N,out);
[res(:,:,5),res(:,:,6)] = addtominus(N,out);
[res(:,:,7),res(:,:,8)] = posmov(N,out,low,high);
[res(:,:,9),res(:,:,10)] = freqmov(N,out,low,high);
[res(:,:,11),res(:,:,12)] = cycconv(N,out,low,high);
for n = 1:6
    str = chstr{ n };
    if abs(res(:,:,2* (n - 1) + 1) - res(:,:,2* n)) < preci
       disp(strcat(str,' is TRUE'));
    else
       disp(strcat(str,' is NOT True'));
    end
end
function [FmG,FG] = linearXZ(N,fbase,preci)
f = rand(N); g = rand(N); a = rand(1,1); b = rand(1,1);
fg = a* f + b* g;
F = fbase* (a* f)* fbase/N^2; G = fbase* (b* g)* fbase/N^2;
FG = fbase* fg* fbase/N^2; FmG = F + G;
function [FF,G] = rotate180(N,fbase)
f = rand(N); F = fbase* (f)* fbase/N^2;
FF = fbase* (F)* fbase; G = fliplr(flipud(f));
function [p2mF,T] = addtominus(N,fbase)
f = rand(N); p2mf = fliplr(flipud(f)); F = fbase* (f)* fbase/N^2;
p2mF = fbase* (p2mf)* fbase/N^2; T = fliplr(flipud(F));
function [FF,G] = posmov(N,fbase,low,high)
f = rand(N); r = [1:N]; c = r; rn = 2; cn = 3;
```

```
r=mod(r-rn,N);r(r==0)=N;c=mod(c-cn,N);c(c==0)=N;
for m=1:N
    for n=1:N
       fmvd(m,n)=f(c(m),r(n));
    end
end
FF=fbase*fmvd*fbase/N^2;G=fbase*f*fbase/N^2;
U=repmat([-low:high],N,1);V=repmat([-low:high]',1,N);
G=G.*exp(-j*2*pi*(U*rn+V*cn)/N);
function [FF,G]=freqmov(N,fbase,low,high)
f=rand(N);F=fbase*f*fbase/N^2;
r=[1:N];c=r;rn=1;cn=1;
r=mod(r-rn,N);r(r==0)=N;
c=mod(c-cn,N);c(c==0)=N;
for m=1:N
    for n=1:N
        G(m,n)=F(c(m),r(n));
    end
end
X=repmat([-low:high],N,1);Y=repmat([-low:high]',1,N);
fmvd=f.*exp(j*2*pi*(X*rn+Y*cn)/N);
FF=fbase*fmvd*fbase/N^2;
function [FF,G]=cycconv(N,fbase,low,high)
f=rand(N);g=randn(N);gminus=fliplr(flipud(g));gmvd=zeros(N,N);
for x=-low:high
    xcorr=mod([1:N]-x+N,N);xcorr(xcorr==0)=N;
    for y=-low:high
        ycorr=mod([1:N]-y+N,N);ycorr(ycorr==0)=N;
        for r=1:N
            for c=1:N
                gmvd(r,c)=gminus(ycorr(r),xcorr(c));
            end
        end
        fg(y+low+1,x+low+1)=sum(sum(f.*gmvd));
    end
end
FF=fbase*fg*fbase/N^2;F=fbase*f*fbase/N^2;
X=fbase*g*fbase;G=F.*X;
```

5.2.2　二维沃尔什和哈达玛变换

对于一个 $N\times N$ 元素组成的图像信号矩阵 X，其二维沃尔什变换定义为

$$T_W = W_N X W_N \tag{5-49}$$

反变换基与正变换基相同。沃尔什基矩阵与一维变换时相同，由沃尔什函数产生。由沃尔什矩阵可以构成二维沃尔什函数的基本图像。具体方法为：以基矩阵 W_N 的各行作为图像的一条边，以矩阵的各列作为图像的另一边，分别相乘即可得到 N^2 的沃尔什基本图像。如以 $N=4$ 为例，4 阶沃尔什基本图像 X 轴方向的 4 条边分别为［1 1 1 1］，［1 1 −1 −1］，［1 −1 −1 1］，［1 −1 1 −1］；因为基矩阵具有主对称性质，所以 Y 轴方向 4 条边与 X 轴相同，相乘即得到 16 个沃尔什基本图像，对应矩阵见表 5-2。

表 5-2　4 阶沃尔什基本图像矩阵

	1	1	1	1	1	1	−1	−1	1	−1	−1	1	1	−1	1	−1
1	1	1	1	1	1	1	−1	−1	1	−1	−1	1	1	−1	1	−1
1	1	1	1	1	1	1	−1	−1	1	−1	−1	1	1	−1	1	−1
1	1	1	1	1	1	1	−1	−1	1	−1	−1	1	1	−1	1	−1
1	1	1	1	1	1	1	−1	−1	1	−1	−1	1	1	−1	1	−1
	W (0, 0)				***W*** (1, 0)				***W*** (2, 0)				***W*** (3, 0)			
1	1	1	1	1	1	1	−1	−1	1	−1	−1	1	1	−1	1	−1
1	1	1	1	1	1	1	−1	−1	1	−1	−1	1	1	−1	1	−1
−1	−1	−1	−1	−1	−1	−1	1	1	−1	1	1	−1	−1	1	−1	1
−1	−1	−1	−1	−1	−1	−1	1	1	−1	1	1	−1	−1	1	−1	1
	W (0, 1)				***W*** (1, 1)				***W*** (2, 1)				***W*** (3, 1)			
1	1	1	1	1	1	1	−1	−1	1	−1	−1	1	1	−1	1	−1
−1	−1	−1	−1	−1	−1	−1	1	1	−1	1	1	−1	−1	1	−1	1
−1	−1	−1	−1	−1	−1	−1	1	1	−1	1	1	−1	−1	1	−1	1
1	1	1	1	1	1	1	−1	−1	1	−1	−1	1	1	−1	1	−1
	W (0, 2)				***W*** (1, 2)				***W*** (2, 2)				***W*** (3, 2)			
1	1	1	1	1	1	1	−1	−1	1	−1	−1	1	1	−1	1	−1
−1	−1	−1	−1	−1	−1	−1	1	1	−1	1	1	−1	−1	1	−1	1
1	1	1	1	1	1	1	−1	−1	1	−1	−1	1	1	−1	1	−1
−1	−1	−1	−1	−1	−1	−1	1	1	−1	1	1	−1	−1	1	−1	1
	W (0, 3)				***W*** (1, 3)				***W*** (2, 3)				***W*** (3, 3)			

与傅里叶基本图像不同的是，沃尔什变换等其他正交变换基图像均是实矩阵。构造基本图像及其列率谱的 MATLAB 实现代码（只列出基矩阵生成过程）如下：

```
function out =wal2Fun(N)
if nargin = =0
  path(path,'d:/work/book1/chapter 4');
  N =4;wbase =walFun(N),M =64/N;
end
for v =1:N
    for u =1:N
        n = (v -1)* N +u;
        wbi(:,:,n) =wbase(:,v)* wbase(u,:);
        tw(:,:,n) =wbase* wbi(:,:,n)* wbase/N ^ 2;
        wbi(:,:,n) = (wbi(:,:,n) +1)/2;
    end
end
```

运行结果如图 5-5 所示。

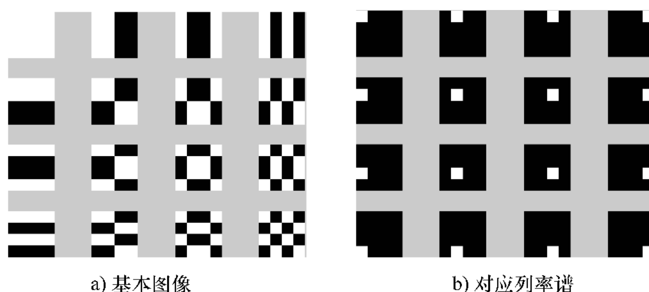


图 5-5　基本图像及列率谱

任意一幅 $N \times N$ 图像都可以分解成 N^2 个基本图像加权和，权系数即为列率谱矩阵 $\boldsymbol{T}_{\mathrm{W}}$ 对应元素值。二维沃尔什变换具有以下性质：

（1）线性

令 $\mathscr{W}\{f(x, y)\}$ 表示离散函数 $f(x, y)$ 的傅里叶变换，即 $T(u,v) = \mathscr{W}\{f(x,y)\}$，则式（5-36）定义的傅里叶变换可得：

$$\begin{aligned}\mathscr{W}\{af_1(x, y) + bf_2(x, y)\} &= a\,\mathscr{W}\{af_1(x, y)\} + b\,\mathscr{W}\{f_2(x, y)\} \\ &= aT_1(u, v) + bT_2(u, v)\end{aligned} \tag{5-50}$$

（2）列率搬移

两个二维沃尔什函数相乘为

$$\mathrm{Wal}(k, m, x, y)\cdot\mathrm{Wal}(l, n, x, y) = \mathrm{Wal}(k \otimes l, m \otimes n, x, y) \tag{5-51}$$

（3）并元移位

如有原函数 $f(x, y)$，其沃尔什变换为 $F(k, m)$，自变量 x、y 做 s、t 位并元移位成为 $f(x \otimes s, y \otimes t)$ 的沃尔什变换为 $F(k, m)_{s,t}$，则有以下关系成立：

$$F(k,m)_{s,t} = F(k,m) \cdot \mathrm{Wal}(k,m,s,t) \tag{5-52}$$

（4）循环并元卷积

信号的二维并元卷积的沃尔什变换等于信号沃尔什变换乘积，即

$$\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)g(x \otimes s, y \otimes t) \to F(k,m)G(k,m) \tag{5-53}$$

（5）调制性质

信号乘积的沃尔什变换等于信号沃尔什变换的并元卷积，即

$$f(x,y)g(x,y) \to \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} F(k,m)G(k \otimes l, m \otimes n) \tag{5-54}$$

用 MATLAB 验证上述性质的代码如下：

```
function out = walXZ(N)
if nargin = =0
    path(path,'d:/work/book1/chapter 4');N =4;wbase =walFun(N);
    preci =exp( -10);
    chstr ={ 'linearity','seqmov','bymov','conv','mod' };
end
len =length(chstr);res =zeros(N,N,2* len);
[res(:,:,1),res(:,:,2)] =linearXZ(N,wbase);
[res(:,:,3),res(:,:,4)] =seqmovXZ(N,wbase);
[res(:,:,5),res(:,:,6)] =bymovXZ(N,wbase);
[res(:,:,7),res(:,:,8)] =convXZ(N,wbase);
[res(:,:,9),res(:,:,10)] =modXZ(N,wbase);
for n =1:len
    str =chstr{ n };
    if abs(res(:,:,2* (n -1) +1) -res(:,:,2* n)) <preci
      disp(strcat(str,' is TRUE'));
    else
      disp(strcat(str,' is NOT True'));
    end
end
function [FmG,FG] =linearXZ(N,fbase)
f =rand(N);g =rand(N);a =rand(1,1);b =rand(1,1);
fg =a* f +b* g;
F =fbase* (a* f)* fbase;G =fbase* (b* g)* fbase;
FG =fbase* fg* fbase;FmG =F +G;
function [FF,G] =seqmovXZ(N,fbase)
OUT =floor(rand(1,4)* (N -1)) +1;
L =OUT(3);N =OUT(4);K =OUT(1);M =OUT(2);
KL =bitxor(K -1,L -1) +1;MN =bitxor(M -1,N -1) +1;
WalKM =fbase(:,M)* fbase(K,:);
```

```
WalLN = fbase(:,N)* fbase(L,:);
FF = WalKM.* WalLN;G = fbase(:,MN)* fbase(KL,:);
function [FF,G] = bymovXZ(N,fbase)
f = rand(N);FF = fbase* f* fbase;
OUT = floor(rand(1,2)* (N -1));S = OUT(1);T = OUT(2);
for x = 1:N
    for y = 1:N
      xby = bitxor(x -1,S) +1;yby = bitxor(y -1,T) +1;
      fby(yby,xby) = f(y,x);
    end
end
G = fbase* fby* fbase;FF = FF.* (fbase(:,T +1)* fbase(S +1,:));
function [FF,FG] = convXZ(N,fbase)
f = rand(N);F = fbase* f* fbase;g = rand(N);G = fbase* g* fbase;
FG = F.* G;
for s = 0:N -1
    for t = 0:N -1
        for x = 1:N
            for y = 1:N
                xby = bitxor(x -1,s) +1;yby = bitxor(y -1,t) +1;
                gby(yby,xby) = g(y,x);
            end
        end
        fg(t +1,s +1) = sum(sum(gby.* f));
    end
end
FF = fbase* fg* fbase;
function [FG,FBG] = modXZ(N,fbase,low,high)
f = rand(N);F = fbase* f* fbase;g = rand(N);G = fbase* g* fbase;
fg = f.* g;FG = fbase* fg* fbase;
for s = 0:N -1
    for t = 0:N -1
        for x = 1:N
            for y = 1:N
                xby = bitxor(x -1,s) +1;yby = bitxor(y -1,t) +1;
                gby(yby,xby) = G(y,x);
            end
        end
        FBG(t +1,s +1) = sum(sum(gby.* F))/N^2;
    end
end
```

二维哈达玛变换基矩阵与沃尔什基矩阵只是行向量排列顺序不同，正反变换形式相同。相应的，二维哈达玛变换基图像与沃尔什变换相同，只是顺序不同。

5.2.3　二维正弦变换

离散余弦变换经常被信号处理和图像处理使用，用于对信号和图像（包括静止图像和运动图像）进行有损数据压缩。这是由于离散余弦变换具有很强的“能量集中”特性：大多数的自然信号（包括声音和图像）的能量都集中在离散余弦变换后的低频部分，而且当信号具有接近马尔可夫过程的统计特性时，离散余弦变换的去相关性接近于 K-L 变换性能。对于任一图像 $\boldsymbol{X}$ 的二维余弦变换和逆变换可表示如下：

$$\begin{aligned} \boldsymbol{T}_c &= \boldsymbol{C}\boldsymbol{X}\boldsymbol{C}^{\mathrm{T}} \\ \boldsymbol{X} &= \boldsymbol{C}^{\mathrm{T}}\boldsymbol{T}_c\boldsymbol{C} \end{aligned} \tag{5-55}$$

二维余弦变换的基本图像类似傅里叶基本图像构成，如 $N=4$ 时 16 个基矩阵见表 5-3。

表 5-3　4 阶余弦基本图像矩阵

	0.5	0.5	0.5	0.5	0.65	0.27	-0.27	-0.65	0.5	-0.5	-0.5	0.5	0.27	-0.65	0.65	-0.27
0.5	0.25	0.25	0.25	0.25	0.33	0.14	-0.14	-0.33	0.25	-0.25	-0.25	0.25	0.14	-0.33	0.33	-0.14
0.5	0.25	0.25	0.25	0.25	0.33	0.14	-0.14	-0.33	0.25	-0.25	-0.25	0.25	0.14	-0.33	0.33	-0.14
0.5	0.25	0.25	0.25	0.25	0.33	0.14	-0.14	-0.33	0.25	-0.25	-0.25	0.25	0.14	-0.33	0.33	-0.14
0.5	0.25	0.25	0.25	0.25	0.33	0.14	-0.14	-0.33	0.25	-0.25	-0.25	0.25	0.14	-0.33	0.33	-0.14
	C（0，0）				**C**（1，0）				**C**（2，0）				**C**（3，0）			
0.65	0.33	0.33	0.33	0.33	0.43	0.18	-0.18	-0.43	0.33	-0.33	-0.33	0.33	0.18	-0.43	0.43	-0.18
0.27	0.14	0.14	0.14	0.14	0.18	0.07	-0.07	-0.18	0.14	-0.14	-0.14	0.14	0.07	-0.18	0.18	-0.07
-0.27	-0.14	-0.14	-0.14	-0.14	-0.18	-0.07	0.07	0.18	-0.14	0.14	0.14	-0.14	-0.07	0.18	-0.18	0.08
-0.65	-0.33	-0.33	-0.33	-0.33	-0.43	-0.18	0.18	0.43	-0.33	0.33	0.33	-0.33	-0.18	0.43	-0.43	0.18
	C（0，1）				**C**（1，1）				**C**（2，1）				**C**（3，1）			
0.5	0.25	0.25	0.25	0.25	0.33	0.14	-0.14	-0.33	0.25	-0.25	-0.25	0.25	0.14	-0.33	0.33	-0.14
-0.5	-0.25	-0.25	-0.25	-0.25	-0.33	-0.14	0.14	0.33	-0.25	0.25	0.25	-0.25	-0.14	0.33	-0.33	0.14
-0.5	-0.25	-0.25	-0.25	-0.25	-0.33	-0.14	0.14	0.33	-0.25	0.25	0.25	-0.25	-0.14	0.33	-0.33	0.14
0.5	0.25	0.25	0.25	0.25	0.33	0.14	-0.14	-0.33	0.25	-0.25	-0.25	0.25	0.14	-0.33	0.33	-0.14
	C（0，2）				**C**（1，2）				**C**（2，2）				**C**（3，2）			
0.27	0.14	0.14	0.14	0.14	0.18	0.07	-0.07	-0.18	0.14	-0.14	-0.14	0.14	0.07	-0.18	0.18	-0.07
-0.65	-0.33	-0.33	-0.33	-0.33	-0.43	-0.18	0.18	0.43	-0.33	0.33	0.33	-0.33	-0.18	0.43	-0.43	0.18
0.65	0.33	0.33	0.33	0.33	0.43	0.18	-0.18	-0.43	0.33	-0.33	-0.33	0.33	0.18	-0.43	0.43	-0.18
-0.27	-0.14	-0.14	-0.14	-0.14	-0.18	-0.07	0.07	0.18	-0.14	0.14	0.14	-0.14	-0.07	0.18	-0.18	0.08
	C（0，3）				**C**（1，3）				**C**（2，3）				**C**（3，3）			

用 MATLAB 构造余弦基本图像及其谱的实现代码（只列出基矩阵生成过程）如下：

```
function out = cos2Fun(N)
if nargin = =0
  path(path,'d:/work/book1/chapter 4');
  N =4;cbase =cosFun(N);M =64/N;cbaset =cbase';
  minc =min(min(cbase));maxc =max(max(cbase));
  mincc =maxc* minc;maxcc =maxc* maxc;cscope =maxcc -mincc;
end
for v =1:N
    for u =1:N
        n = (v -1)* N +u;
        cbi(:,:,n) =cbaset(:,v)* cbase(u,:);
        tw(:,:,n) =cbase* cbi(:,:,n)* cbaset;
        cbi(:,:,n) = (cbi(:,:,n)   -mincc)/cscope;
    end
end
```

运行结果如图 5-6 所示。

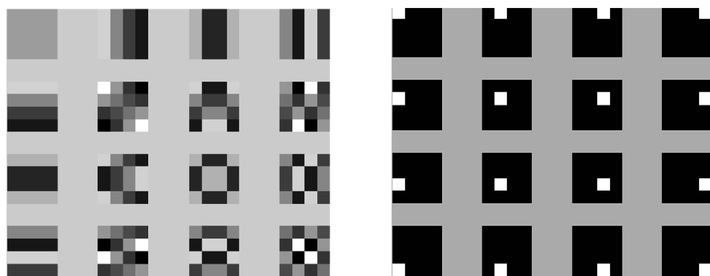


图 5-6　基本图像及频谱

5.3　二维全相位信号处理

参考图 4-12，假设变换域系统 $\boldsymbol{H}$ 是 $N\times N$ 维的，对于任意元素 x_{ij} 在原点位置时对应的输入矩阵 $\boldsymbol{X}^{(0,0)}$ 定义如下：

$$\boldsymbol{X}^{(0,0)}=\begin{bmatrix} x_{i,j} & x_{i,j+1} & \cdots & x_{i,j+N-1} \\ x_{i+1,j} & x_{i+1,j+1} & \cdots & x_{i+1,j+N-1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{i+N-1,j} & x_{i+N-1,j+1} & \cdots & x_{i+N-1,j+N-1} \end{bmatrix}_{N\times N} \tag{5-56}$$

传输特性序列 $\boldsymbol{H}$ 定义如下：

$$\boldsymbol{H}=\begin{bmatrix} H_{0,0} & H_{0,1} & \cdots & H_{0,N-1} \\ H_{1,0} & H_{1,1} & \cdots & H_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ H_{N-1,0} & H_{N-1,1} & \cdots & H_{N-1,N-1} \end{bmatrix}_{N\times N} \tag{5-57}$$

由上面 5.1 节介绍的正交变换可得输出 $\boldsymbol{Y}$ 为：

$$\boldsymbol{Y}=\boldsymbol{\beta}[(\boldsymbol{\alpha}\boldsymbol{X}^{(0,0)}\boldsymbol{\alpha}^{\mathrm{T}})\otimes\boldsymbol{H}]\boldsymbol{\beta}^{\mathrm{T}} \tag{5-58}$$

结合图 4-12，有

$$\boldsymbol{I}_{m,n}=\sum_{p=0}^{N-1}\sum_{q=0}^{N-1}\boldsymbol{\alpha}_{m,p}\boldsymbol{\alpha}_{n,q}\boldsymbol{X}_{p,q}^{(0,0)} \tag{5-59}$$

$$\boldsymbol{O}_{m,n}=\boldsymbol{I}_{m,n}\boldsymbol{H}_{m,n}=\sum_{p=0}^{N-1}\sum_{q=0}^{N-1}\boldsymbol{\alpha}_{m,p}\boldsymbol{\alpha}_{n,q}\boldsymbol{H}_{m,n}\boldsymbol{X}_{p,q}^{(0,0)} \tag{5-60}$$

$$\begin{aligned}\boldsymbol{Y}^{(0,0)}&=\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\boldsymbol{\beta}_{i,m}\boldsymbol{\beta}_{j,n}\boldsymbol{O}_{m,n}\\&=\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\boldsymbol{\beta}_{0,m}\boldsymbol{\beta}_{0,n}\sum_{p=0}^{N-1}\sum_{q=0}^{N-1}\boldsymbol{\alpha}_{m,p}\boldsymbol{\alpha}_{n,q}\boldsymbol{H}_{m,n}\boldsymbol{X}_{p,q}^{(0,0)}\\&=\sum_{p=0}^{N-1}\sum_{q=0}^{N-1}\left(\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\boldsymbol{\alpha}_{m,p}\boldsymbol{\alpha}_{n,q}\boldsymbol{\beta}_{0,m}\boldsymbol{\beta}_{0,n}\boldsymbol{H}_{m,n}\right)\boldsymbol{X}_{p,q}^{(0,0)}\end{aligned} \tag{5-61}$$

对于输入 $\boldsymbol{X}^{(0,0)}$，对应 $x_{i,j}$的输出是（$u=0$，$v=0$）位置的 $Y_{0,0}{}^{(0,0)}$。二维全相位数字处理的输入是 $x_{i,j}$遍历 $N\times N$ 个位置的 N^2个矩阵，输出则是对应 $\boldsymbol{X}^{(u,v)}$ 在位置（u，v）处 N^2个输出 $Y_{u,v}{}^{(u,v)}$的和的平均。由式（5-56）可知遍历矩阵 $\boldsymbol{X}^{(u,v)}$为

$$\boldsymbol{X}^{(u,v)}=\begin{bmatrix} x_{i-u,j-v} & x_{i-u,j-v+1} & \cdots & x_{i-u,j-v+N-1} \\ x_{i-u+1,j-v} & x_{i-u+1,j-v+1} & \cdots & x_{i-u+1,j-v+N-1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{i-u+N-1,j-v} & x_{i-u+N-1,j-v+1} & \cdots & x_{i-u+N-1,j-v+N-1} \end{bmatrix}_{N\times N} \tag{5-62}$$

结合式（5-61）和式（5-62），可知二维全相位信号处理输出为

$$y=\frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}Y_{u,v}=\frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\left[\sum_{p=0}^{N-1}\sum_{q=0}^{N-1}\left(\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\boldsymbol{\alpha}_{m,p}\boldsymbol{\alpha}_{n,q}\boldsymbol{\beta}_{u,m}\boldsymbol{\beta}_{v,n}\boldsymbol{H}_{m,n}\right)\boldsymbol{X}_{p,q}^{(u,v)}\right] \tag{5-63}$$

观测式（5-62）易知：

$$\boldsymbol{X}_{p,q}^{(u,v)}=x_{i+p-u,j+q-v} \tag{5-64}$$

令

$$\boldsymbol{T}_{p,q}^{(u,v)}=\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\boldsymbol{\alpha}_{m,p}\boldsymbol{\alpha}_{n,q}\boldsymbol{\beta}_{u,m}\boldsymbol{\beta}_{v,n}\boldsymbol{H}_{m,n} \tag{5-65}$$

把式（5-64）和式（5-65）代入到式（5-63）得：

$$y = \frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\left[\sum_{p=0}^{N-1}\sum_{q=0}^{N-1}\boldsymbol{T}_{p,q}^{(u,v)}x_{i+p-u,j+q-v}\right]$$
$$\overset{r=p-u,\ c=q-v}{=} \frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\left[\sum_{r=-u}^{N-1-u}\sum_{c=-v}^{N-1-v}\boldsymbol{T}_{r+u,c+v}^{(u,v)}x_{i+r,j+c}\right] \tag{5-66}$$

分别交换 r 与 x 和 c 与 y 的求和顺序后得：

$$y = \frac{1}{N^2}\left(\sum_{r=1}^{N-1}\sum_{u=0}^{N-1-r} + \sum_{r=-N+1}^{0}\sum_{u=-r}^{N-1}\right)\left(\sum_{c=1}^{N-1}\sum_{v=0}^{N-1-c} + \sum_{c=-N+1}^{0}\sum_{v=-c}^{N-1}\right)\boldsymbol{T}_{r+u,c+v}^{(u,v)}x_{i+r,j+c}$$
$$= \frac{1}{N^2}\left(\sum_{r=1}^{N-1}\sum_{u=0}^{N-1-r}\sum_{c=1}^{N-1}\sum_{v=0}^{N-1-c} + \sum_{r=1}^{N-1}\sum_{u=0}^{N-1-r}\sum_{c=-N+1}^{0}\sum_{v=-c}^{N-1} + \sum_{r=-N+1}^{0}\sum_{u=-r}^{N-1}\sum_{c=1}^{N-1}\sum_{v=0}^{N-1-c} + \sum_{r=-N+1}^{0}\sum_{u=-r}^{N-1}\sum_{c=-N+1}^{0}\sum_{v=-c}^{N-1}\right)\boldsymbol{T}_{r+u,c+v}^{(u,v)}x_{i+r,j+c}$$
$$= \frac{1}{N^2}\left[\sum_{r=1}^{N-1}\sum_{c=1}^{N-1}\left(\sum_{u=0}^{N-1-r}\sum_{v=0}^{N-1-c}\boldsymbol{T}_{r+u,c+v}^{(u,v)}\right) + \sum_{r=1}^{N-1}\sum_{c=-N+1}^{0}\left(\sum_{u=0}^{N-1-r}\sum_{v=-c}^{N-1}\boldsymbol{T}_{r+u,c+v}^{(u,v)}\right) + \sum_{r=-N+1}^{0}\sum_{c=1}^{N-1}\left(\sum_{u=-r}^{N-1}\sum_{v=0}^{N-1-c}\boldsymbol{T}_{r+u,c+v}^{(u,v)}\right) + \sum_{r=-N+1}^{0}\sum_{c=-N+1}^{0}\left(\sum_{u=-r}^{N-1}\sum_{v=-c}^{N-1}\boldsymbol{T}_{r+u,c+v}^{(u,v)}\right)\right]x_{i+r,j+c} \tag{5-67}$$

观察式（5-65），矩阵 $\boldsymbol{T}$ 是由正交变换正反核与特性矩阵共同定义，它是数据 $x_{i,j}$ 移动到（u，v）时对应输入 $\boldsymbol{X}$ 的加权，称 $\boldsymbol{T}$ 为二维全相位变换核。观察式（5-67），对于不同区间的（r，c），定义变量如下：

$$\boldsymbol{A}_{r,c} = \begin{cases} \boldsymbol{A}^{++} = \left[\dfrac{1}{N^2}\sum_{u=0}^{N-1-r}\sum_{v=0}^{N-1-c}T_{r+u,c+v}^{(u,v)}\right] & 1 \leqslant r \leqslant N-1,\ 1 \leqslant c \leqslant N-1 \\ \boldsymbol{A}^{+-} = \left[\dfrac{1}{N^2}\sum_{u=0}^{N-1-r}\sum_{v=-c}^{N-1}\boldsymbol{T}_{r+u,c+v}^{(u,v)}\right] & 1 \leqslant r \leqslant N-1,\ -N+1 \leqslant c \leqslant 0 \\ \boldsymbol{A}^{-+} = \left[\dfrac{1}{N^2}\sum_{u=-r}^{N-1}\sum_{v=0}^{N-1-c}\boldsymbol{T}_{r+u,c+v}^{(u,v)}\right] & -N+1 \leqslant r \leqslant 0,\ 1 \leqslant c \leqslant N-1 \\ \boldsymbol{A}^{--} = \left[\dfrac{1}{N^2}\sum_{u=-r}^{N-1}\sum_{v=-c}^{N-1}\boldsymbol{T}_{r+u,c+v}^{(u,v)}\right] & -N+1 \leqslant r \leqslant 0,\ -N+1 \leqslant c \leqslant 0 \end{cases} \tag{5-68}$$

由式（5-68）可以看出，矩阵 $\boldsymbol{A}$ 是通过对不同区间的全相位变换核求和得到，它包含四个子矩阵即 $\boldsymbol{A}^{++}$、$\boldsymbol{A}^{+-}$、$\boldsymbol{A}^{-+}$ 和 $\boldsymbol{A}^{--}$，其物理意义是数据 $x_{i,j}$ 由（0，0）位置逐点移动到（$N-1$，$N-1$）过程中，以数据 $x_{i,j}$ 划分的 $\boldsymbol{T}$ 的 4 个区间内各自数据之和即得到 $\boldsymbol{A}$ 的 4 个子矩阵，形成过程如图 5-7 所示。

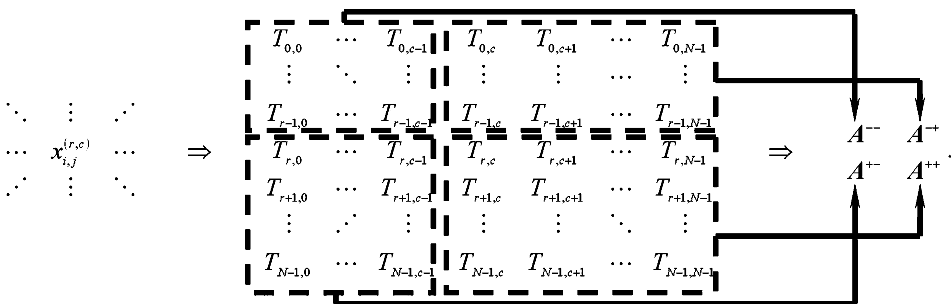


图 5-7　矩阵 $\boldsymbol{A}$ 形成过程

由图 5-7 可以看出，四个子矩阵 $\boldsymbol{A}^{++}$、$\boldsymbol{A}^{+-}$、$\boldsymbol{A}^{-+}$和 $\boldsymbol{A}^{--}$的维数分别是 $N\times N$、$N\times(N-1)$、$(N-1)\times N$ 和$(N-1)\times(N-1)$，所以矩阵 $\boldsymbol{A}$ 的维数是$(2N-1)\times(2N-1)$。把式（5-68）代入到式（5-67）得：

$$\begin{aligned} y &= \left[\sum_{r=1}^{N-1}\sum_{c=1}^{N-1} + \sum_{r=1}^{N-1}\sum_{c=-N+1}^{0} + \sum_{r=-N+1}^{0}\sum_{c=1}^{N-1} + \sum_{r=-N+1}^{0}\sum_{c=-N+1}^{0}\right]\boldsymbol{A}_{r,c}x_{i+r,j+c} \\ &= \sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1}\boldsymbol{A}_{r,c}x_{i+r,j+c} \end{aligned} \tag{5-69}$$

式（5-69）表明，二维全相位信号处理对应 $x_{i,j}$的输出等于在行列方向上以 $x_{i,j}$为中心的共（$2N-1$）^{2}个数据以 $\boldsymbol{A}$ 为加权的总和。式（5-69）称为二维全相位变换，$\boldsymbol{A}$ 是全相位变换基矩阵。

参考图 5-1 的信号移动规律，二维全相位信号处理可以按照行或者列优先的方式来实现，下面以行优先为例给出实现原理图。行方向上块处理方式如图 5-8 所示。

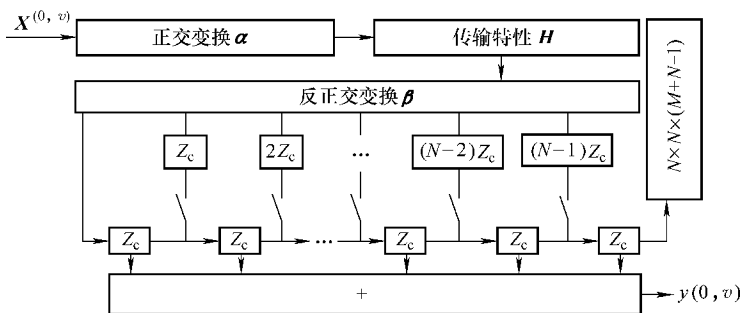


图 5-8　二维全相位行信号处理框图

图 5-8 中，输入是按照行方向移动的大小为 $N\times N$ 的数据块，假设待处理图像大小为 $M\times L$，且进行边界扩展成$(M+2N-1)\times(L+2N-1)$以保证输出图像大小不变。Z_c是行方向移动的单位延迟。图 5-8 工作方式是，按照间隔 Z_c顺序输入的 N 个起始方块经过正交变换、$\boldsymbol{H}$ 加权和反变换后依次进入延迟分别为 iZ_c（$i=N-1$，$N-2$，…，0）的缓存器中，$\boldsymbol{X}^{(0,N-1)}$处理后的 N 个数据块同时进入加法器

上方的单位延迟缓存器，同时断开各路开关。后续的 $\boldsymbol{X}^{(0,N)}$ 即对应 $x_{i,j+1}$ 的输入处理结果将直接进入最左边直通通路而顺序产生 M 个对应输出 $y(0, v)$，并把 $M+N-1$ 个处理结果存入 buffer 中。封装图 5-8 成子系统如图 5-9所示。

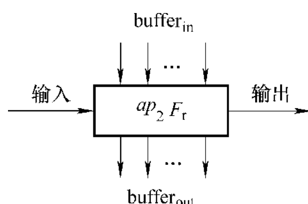


图 5-9　二维全相位行信号处理封装模块

图中 buffer$_{out}$是本模块处理的所有中间结果，buffer$_{in}$是下一行模块处理的所有中间结果作为当前模块处理时初始状态值，存入加法器上方的 N 个单位延迟缓存器中。这样，与 $x_{i,j}$ 相关的 N 行并行处理系统可设计成如图 5-10 所示的结构。

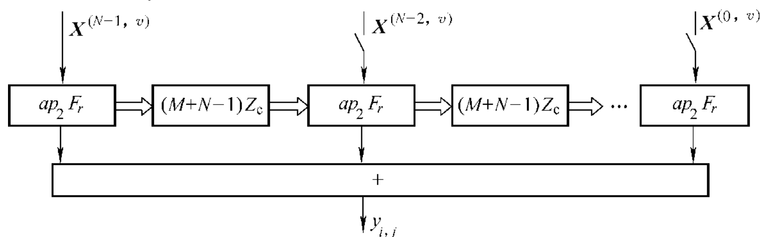


图 5-10　二维全相位信号处理系统图

图 5-10 工作流程是，初始 N 个数据块输入后，断开所有开关。N 列并行处理，经过 $(M+N-1)Z_c$ 时间后，所有行均完成处理并输出 N 个结果，这样对应 $x_{i,j}$ 的 N^2 个数据和即为输出 $y_{i,j}$。以上是按照式（5-63）的实现过程，与其等价的式（5-69）实现方式只需用全相位变换基按照行或列优先逐点移动处理来完成。一种直接方式如图 5-11 所示。

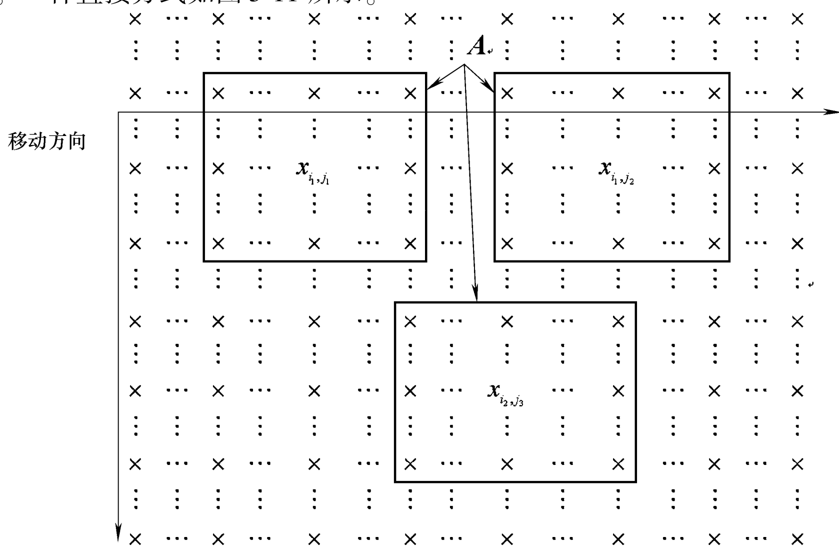


图 5-11　全相位基滑动处理图

由图 5-11 所示的处理过程可以看出，通过滑动全相位基矩阵 $\boldsymbol{A}$ 来对准某像素 $x_{i,j}$后按照式（5-69）即得到对应输出 $y_{i,j}$。考虑到图像信号的空间相关性以及为充分利用相邻行或列的重复部分，可做进一步优化处理。对于同行的相邻点 $x_{i,j}$和 $x_{i,j+1}$，由式（5-69）可知对应输出 $y_{i,j}$和 $y_{i,j+1}$具有如下关系：

$$\Delta y_{i,j}^{(c)} = y_{i,j+1} - y_{i,j} = \sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1}\boldsymbol{A}_{r,c}(x_{i+r,j+1+c} - x_{i+r,j+c}) = \sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1}\boldsymbol{A}_{r,c}\Delta x_{i+r,j+c} \tag{5-70}$$

即相邻像素的全相位处理输出差分正比于对应输入块 $\boldsymbol{X}$ 的差分。由此可推导出同一行的任意点输出为

$$y_{i,j} = y_{i,0} + \sum_{j=1}^{N-1}\Delta y_{i,j}^{(c)} = \sum_{j=1}^{N-1}\sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1}\boldsymbol{A}_{r,c}\Delta x_{i+r,j+c} \tag{5-71}$$

对于同列相邻点 $x_{i,j}$和 $x_{i+1,j}$，由式（5-69）可知对应输出 $y_{i,j}$和 $y_{i+1,j}$具有如下关系：

$$\begin{aligned}
\Delta y_{i,j}^{(r)} &= y_{i+1,j} - y_{i,j} = \sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1}\boldsymbol{A}_{r,c}x_{i+1+r,j+c} - \sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1}\boldsymbol{A}_{r,c}x_{i+r,j+c} \\
&\overset{r'=1+r}{=} \sum_{r'=-N+2}^{N}\sum_{c=-N+1}^{N-1}\boldsymbol{A}_{r'-1,c}x_{i+r',j+c} - \sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1}\boldsymbol{A}_{r,c}x_{i+r,j+c} \\
&\overset{r=r'}{=} \sum_{c=-N+1}^{N-1}(\boldsymbol{A}_{N-1,c}x_{i+N,j+c} - \boldsymbol{A}_{-N+1,c}x_{i-N+1,j+c}) + \sum_{r=-N+2}^{N-1}\sum_{c=-N+1}^{N-1}(\boldsymbol{A}_{r-1,c} - \boldsymbol{A}_{r,c})x_{i+r,j+c} \\
\Rightarrow \quad y_{i+1,j} &= y_{i,j} + \sum_{c=-N+1}^{N-1}(\boldsymbol{A}_{N-1,c}x_{i+N,j+c} - \boldsymbol{A}_{-N+1,c}x_{i-N+1,j+c}) - \sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1}\Delta\boldsymbol{A}_{r,c}x_{i+r,j+c} \\
&= \sum_{c=-N+1}^{N-1}\boldsymbol{A}_{N-1,c}x_{i+N,j+c} + \sum_{r=-N+2}^{N-1}\sum_{c=-N+1}^{N-1}\boldsymbol{A}_{r,c}x_{i+r,j+c} - \sum_{r=-N+2}^{N-1}\sum_{c=-N+1}^{N-1}\Delta\boldsymbol{A}_{r,c}x_{i+r,j+c} \\
&\overset{A_{N-1,c}=\Delta A_{N,c}}{=} \sum_{r=-N+2}^{N}\sum_{c=-N+1}^{N-1}\Delta\boldsymbol{A}_{r,c}x_{i+r,j+c} + \sum_{r=-N+2}^{N-1}\sum_{c=-N+1}^{N-1}\boldsymbol{A}_{r,c}x_{i+r,j+c}
\end{aligned} \tag{5-72}$$

式（5-72）表明，隔行的输出由两部分组成，第一部分是新增行与全相位基行差分与 $\boldsymbol{X}^{(i+1,j)}$ 的加权和，第二部分是上一行元素全相位处理中全相位基与 $\boldsymbol{X}^{(i,j)}$ 的加权和中除去第一行之和。其中第二部分已经在上一行处理中计算出来，因此可重复利用。利用全相位基的对称性可进一步优化计算过程，下面针对 DFT、DWT 和 DCT 三个变换域分别展开讨论。

5.4 二维 DFT 全相位

对于二维傅里叶变换，$\boldsymbol{\alpha} = \boldsymbol{F}_{\mathrm{N}}$，$\boldsymbol{\beta} = \boldsymbol{F}_{\mathrm{N}}^*$，代入到式（5-65）得：

$$\boldsymbol{T}_{p,q}^{(u,v)} = \sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\boldsymbol{F}_{m,p}\boldsymbol{F}_{n,q}\boldsymbol{F}_{u,m}^*\boldsymbol{F}_{v,n}^*\boldsymbol{H}_{m,n} \tag{5-73}$$

注意到，

$$\begin{aligned}\boldsymbol{F}_{m,p} &= \boldsymbol{W}_{\mathrm{N}}^{mp} = (\boldsymbol{W}^*)_{\mathrm{N}}^{-mp} = (\boldsymbol{W}_{\mathrm{N}}^{-mp})^* = \boldsymbol{F}_{m,-p}^*, \boldsymbol{F}_{n,q} = \boldsymbol{W}_{\mathrm{N}}^{nq} = (\boldsymbol{W}^*)_{\mathrm{N}}^{-nq} \\ &= (\boldsymbol{W}_{\mathrm{N}}^{-nq})^* = \boldsymbol{F}_{n,-q}^* \end{aligned} \tag{5-74}$$

把式（5-74）代入到式（5-73）并整理得：

$$\boldsymbol{T}_{p,q}^{(u,v)} = \sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\boldsymbol{F}_{(u-p),m}^*\boldsymbol{H}_{m,n}\boldsymbol{F}_{n,(v-q)}^* = h(u-p,v-q) \tag{5-75}$$

显然 $\boldsymbol{h}$ 以 N 为周期。把式（5-75）结果代入到式（5-68）可得到全相位基的四个子矩阵为

$$\boldsymbol{A}_{r,c} = \begin{cases} \boldsymbol{A}^{++} = \left[\dfrac{1}{N^2}\displaystyle\sum_{u=0}^{N-1-r}\sum_{v=0}^{N-1-c}h(-r,-c)\right] = \left[\dfrac{(N-r)(N-c)}{N^2}h(-r,-c)\right] \\ \quad 1\leqslant r\leqslant N-1,\ 1\leqslant c\leqslant N-1 \\ \boldsymbol{A}^{+-} = \left[\dfrac{1}{N^2}\displaystyle\sum_{u=0}^{N-1-r}\sum_{v=-c}^{N-1}h(-r,-c)\right] = \left[\dfrac{(N-r)(N+c)}{N^2}h(-r,-c)\right] \\ \quad 1\leqslant r\leqslant N-1,\ -N+1\leqslant c\leqslant 0 \\ \boldsymbol{A}^{-+} = \left[\dfrac{1}{N^2}\displaystyle\sum_{u=-r}^{N-1}\sum_{v=0}^{N-1-c}h(-r,-c)\right] = \left[\dfrac{(N+r)(N-c)}{N^2}h(-r,-c)\right] \\ \quad -N+1\leqslant r\leqslant 0,\ 1\leqslant c\leqslant N-1 \\ \boldsymbol{A}^{--} = \left[\dfrac{1}{N^2}\displaystyle\sum_{u=-r}^{N-1}\sum_{v=-c}^{N-1}h(-r,-c)\right] = \left[\dfrac{(N+r)(N+c)}{N^2}h(-r,-c)\right] \\ \quad -N+1\leqslant r\leqslant 0,\ -N+1\leqslant c\leqslant 0 \end{cases} \tag{5-76}$$

按照图 5-7 方式将 4 个子矩阵组合即可以得到全相位基矩阵 $\boldsymbol{A}$。由延拓单位冲激响应矩阵 $\boldsymbol{h}_e$ 和系数矩阵 $\boldsymbol{C}$ 点乘得到，其维数是$(2N-1)\times(2N-1)$。$\boldsymbol{C}$ 的元素定义如下：

$$\boldsymbol{C}_{r,c} = \frac{1}{N^2}(N-|r|)(N-|c|) \qquad -N+1\leqslant r,\ c\leqslant N-1 \tag{5-77}$$

式中，$\boldsymbol{C}$ 具有可分离性即 $\boldsymbol{C}_{r,c} = \boldsymbol{C}_r\cdot\boldsymbol{C}_c$，其中 $\boldsymbol{C}_r$ 和 $\boldsymbol{C}_c$ 是分别由下式定义的列向量和行向量：

$$\boldsymbol{C}_r = \frac{1}{N}(N-|r|),\ \boldsymbol{C}_c = \frac{1}{N}(N-|c|) \qquad -N+1\leqslant r,\ c\leqslant N-1 \tag{5-78}$$

式中，$\boldsymbol{C}_r$ 和 $\boldsymbol{C}_c$ 是三角窗且均可由长度为 N 的矩形窗卷积生成，称 $\boldsymbol{C}$ 为二维卷积窗。

由式（5-76）可看出 $\boldsymbol{h}_e$ 可由 $\boldsymbol{h}$ 通过周期延拓来形成，即首先由 $\boldsymbol{h}$ 进行行列反转后形成 $\boldsymbol{A}^{--}$，然后在行和列上进行长度为 N 的周期延拓分别形成矩阵 $\boldsymbol{A}^{-+}$、$\boldsymbol{A}^{+-}$ 和 $\boldsymbol{A}^{++}$。观察式（5-76），重新定义行列坐标范围 $r,\ c\in[0,\ 2N-1]$，则

全相位基元素等于：

$$A_{r,c}=\frac{1}{N^2}\begin{cases}(r+1)h(N-1-r,\ N-1-c)(c+1) & 0\leqslant r,\ c\leqslant N-1\\(r+1)h(N-1-r,\ 2N-1-c)(2N-1-c) & 0\leqslant r\leqslant N-1,\ N\leqslant c\leqslant 2N-2\\(2N-1-r)h(2N-1-r,\ N-1-c)(c+1) & N\leqslant r\leqslant 2N-2,\ 0\leqslant c\leqslant N-1\\(2N-1-r)h(2N-1-r,\ 2N-1-c)(2N-1-c) & N\leqslant r,\ c\leqslant 2N-2\end{cases}\tag{5-79}$$

观察式（5-79），全相位基元素由三部分相乘得到，即左矩阵、单位冲激响应矩阵 $\boldsymbol{h}$ 和右矩阵。而左矩阵仅在（r，$N-1-r$）（$0\leqslant r\leqslant N-1$）和（$r$，$2N-1-r$）（$N\leqslant r\leqslant 2N-2$）有值（$r+1$）和（$2N-1-r$），其余各点均为 0；右矩阵仅在（$N-1-c$，$c$）（$0\leqslant c\leqslant N-1$）和（$2N-1-c$，$c$）（$N\leqslant c\leqslant 2N-2$）有值（$c+1$）和（$2N-1-c$），其余各点均为 0。显然，左右矩阵互为转置，且左矩阵可定义如下：

$$\boldsymbol{E}=\begin{bmatrix}0 & \cdots & 0 & 1\\0 & \cdots & 2 & 0\\\vdots & \iddots & \vdots & \vdots\\N & \cdots & 0 & 0\\0 & \cdots & 0 & N-1\\0 & \cdots & N-2 & 0\\\vdots & \iddots & \vdots & \vdots\\0 & \cdots & 0 & 0\end{bmatrix}_{(2N-1)\times N}\tag{5-80}$$

则全相位基可由下式求出：

$$\boldsymbol{A}=\boldsymbol{E}\cdot\boldsymbol{h}\cdot\boldsymbol{E}^{\mathrm{T}}\tag{5-81}$$

整理以上，DFT 二维全相位变换的基本步骤如下：

1）由传输矩阵 $\boldsymbol{H}_{N\times N}$ 求出单位冲激响应矩阵 $\boldsymbol{h}$；

2）按照式（5-80）构造矩阵 $\boldsymbol{E}_{(2N-1)\times N}$；

3）按照式（5-81）求出全相位变换基 $\boldsymbol{A}_{(2N-1)\times(2N-1)}$；

4）对 $M\times M$ 二维输入信号 $\boldsymbol{X}$ 进行边界延拓成（$M+2N-2$）×（$M+2N-2$）；

5）按照式（5-69）进行全相位信号处理。

二维全相位信号处理的输入是以信号 $x_{i,j}$ 为中心的信号块，输出是 $y_{i,j}$，由式（5-69）可求出系统传输特性如下：

$$H(\mathrm{j}\omega_r,\ \mathrm{j}\omega_c)=\sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1}\boldsymbol{A}_{r,c}\mathrm{e}^{-\mathrm{j}(r\omega_r+c\omega_c)}\tag{5-82}$$

把式（5-76）代入到式（5-82）后整理如下：

$$H(\mathrm{j}\omega_r,\mathrm{j}\omega_c)=\frac{1}{N^2}\sum_{r=1}^{N-1}\begin{bmatrix}\sum_{c=1}^{N-1}(N-r)(N-c)h_{-r,-c}\mathrm{e}^{-\mathrm{j}c\omega_c}\\+\sum_{c=-N+1}^{-1}(N-r)(N+c)h_{-r,-c}\mathrm{e}^{-\mathrm{j}c\omega_c}\\+N(N-r)h_{-r,0}\end{bmatrix}\mathrm{e}^{-\mathrm{j}r\omega_r}$$

$$+\sum_{r=-N+1}^{0}\begin{bmatrix}\sum_{c=1}^{N-1}(N+r)(N-c)h_{-r,-c}\mathrm{e}^{-\mathrm{j}c\omega_c}\\+\sum_{c=-N+1}^{-1}(N+r)(N+c)h_{-r,-c}\mathrm{e}^{-\mathrm{j}c\omega_c}\\+N(N+r)h_{-r,0}\end{bmatrix}\mathrm{e}^{-\mathrm{j}r\omega_r}$$

$$=h_{0,0}+\frac{1}{N}\sum_{c=1}^{N-1}(N-c)\left[h_{0,c}\mathrm{e}^{\mathrm{j}c\omega_c}+h_{0,-c}\mathrm{e}^{-\mathrm{j}c\omega_c}\right]+ \tag{5-83}$$

$$\frac{1}{N}\sum_{r=1}^{N-1}(N-r)\left[h_{r,0}\mathrm{e}^{\mathrm{j}r\omega_r}+h_{-r,0}\mathrm{e}^{-\mathrm{j}r\omega_r}\right]+$$

$$\frac{1}{N^2}\sum_{r=1}^{N-1}\sum_{c=1}^{N-1}(N-r)(N-c)$$

$$\left\{\begin{bmatrix}h_{-r,-c}\mathrm{e}^{-\mathrm{j}(r\omega_r+c\omega_c)}+\\h_{r,c}\mathrm{e}^{\mathrm{j}(r\omega_r+c\omega_c)}\end{bmatrix}+\left[h_{-r,c}\mathrm{e}^{-\mathrm{j}(r\omega_r-c\omega_c)}+h_{r,-c}\mathrm{e}^{\mathrm{j}(r\omega_r-c\omega_c)}\right]\right\}$$

由式（5-83）可看出，系统具有线性相位的充分条件是

$$h(r,c)=h(-r,-c) \tag{5-84}$$

系统的传输特性如下：

$$H(\mathrm{j}\omega_r,\mathrm{j}\omega_c)=h_{0,0}+\frac{2}{N}\sum_{c=1}^{N-1}(N-c)h_{0,c}\cos(c\omega_c)+$$

$$\frac{2}{N}\sum_{r=1}^{N-1}(N-r)h_{r,0}\cos(r\omega_r)+$$

$$\frac{2}{N^2}\sum_{r=1}^{N-1}\sum_{c=1}^{N-1}(N-r)(N-c)$$

$$\left[h_{r,c}\cos(r\omega_r+c\omega_c)+h_{r,-c}\cos(r\omega_r-c\omega_c)\right] \tag{5-85}$$

式（5-85）表明，在满足式（5-84）的条件下，系统具有严格零相位，且特性曲面具有 2π 周期性，并关于横纵坐标及原点对称。实际上，由式（5-82）易知，只要全相位基 $\boldsymbol{A}$ 具有中心对称性就能实现线性相位，再结合式（5-76）

同样可以得到式（5-84）的结论。容易证明，式（5-84）成立的充要条件是

$$H(u,\ v)=H(N-u,\ N-v) \tag{5-86}$$

如 8 阶低通滤波特性 $\boldsymbol{H}$ 定义如下：

$$\boldsymbol{H}=\begin{bmatrix}1&1&1&0&0&0&1&1\\1&1&1&0&0&0&0&0\\1&1&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\1&0&0&0&0&0&0&1\\1&0&0&0&0&0&1&1\end{bmatrix} \tag{5-87}$$

其全相位处理系统特性与传统分块处理特性对比如图 5-12 所示。

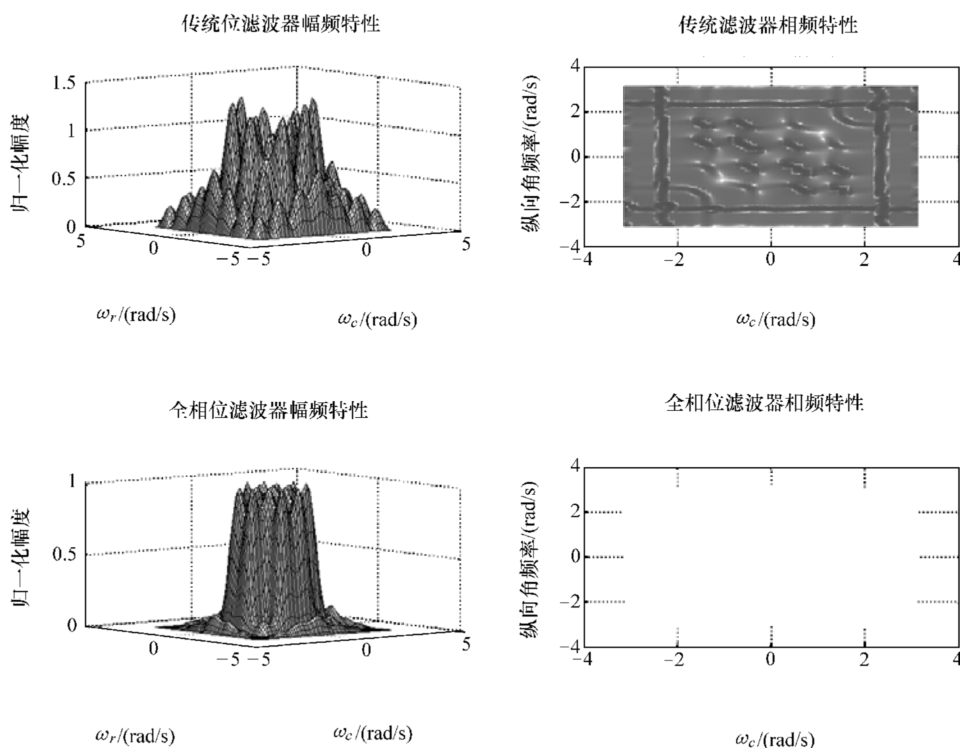


图 5-12　全相位处理与传统处理特性对比

相频特性中纯白色部分代表 0°，纯黑色部分代表 180°，灰色部分代表介于

0°到 180°的相位。从图 5-12 中特性对比可看出，全相位处理幅频特性主瓣能量更加集中且旁瓣分布大幅减少，全相位谱幅值恒为正。为进一步对比全相位处理特性的改善程度，给出了幅频特性 *XZ* 面投影图及 $\omega_r=0$ 时截面图如图 5-13 所示。

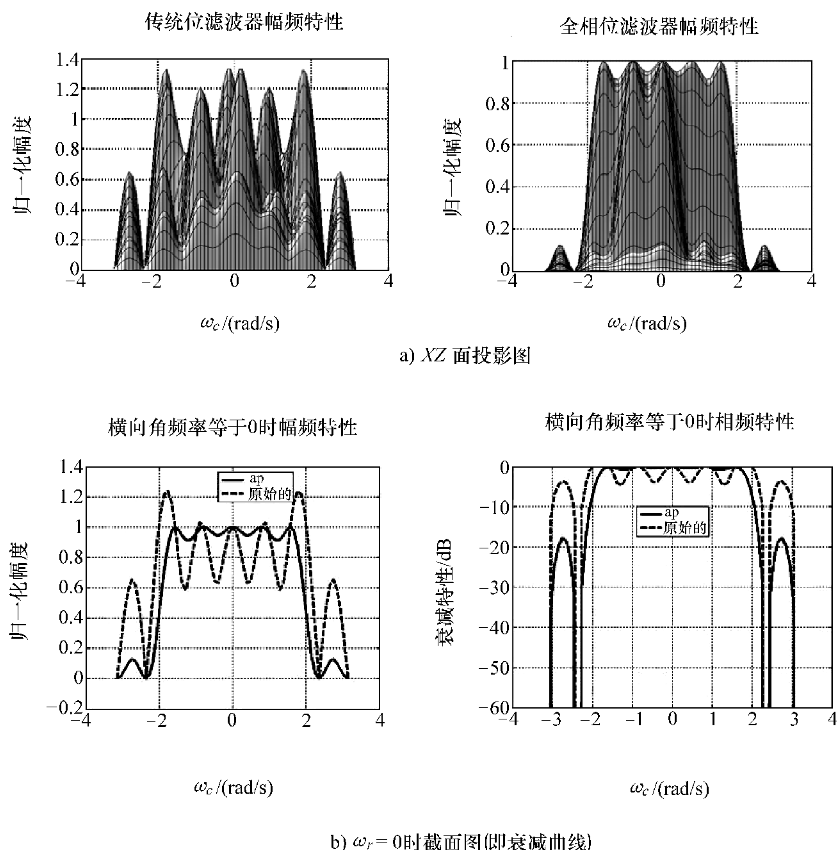


图 5-13　幅频投影和截面对比图

实现图 5-13 所示结果的 MATLAB 代码如下：

```
function out = apDFT2F(N)
if nargin = =0
  N =8;S =3;L =5;cs =L -2;
end
% to build H
H =zeros(N,N);H(1,1) =1;
H(1,2:S) =1;H(1,2:end) =fliplr(H(1,2:end)) +H(1,2:end);
H(2:S,1) =1;H(2:end,1) =flipud(H(2:end,1)) +H(2:end,1);
```

```
for c =2:cs
   for r =2:L -c
      H(r,c) =1;
   end
end
H(2:end,2:end) =H(2:end,2:end) +fliplr(flipud(H(2:end,2:end)));

h =ifft2(H);
wr = -pi:pi/2^5:pi;wc = -pi:pi/2^5:pi;lenw =length(wr);
rs =repmat([0:N -1]',1,N);cs =repmat([0:N -1],N,1);

for r =1:lenw
   for c =1:lenw
HW(r,c) = (sum(sum(exp( -j* rs* wr(r)).* exp( -j* cs* wc(c)).* h)));
p1 =2* sum((N - [1:N -1]).* h(1,2:end).* cos([1:N -1]* wc(c)))/N;
p2 =2* sum((N - [1:N -1]).* h(2:end,1)'.* cos([1:N -1]* wr(r)))/N;
p31 =cos(rs(2:end,2:end)* wr(r) +cs(2:end,2:end)* wc(c)).* h(2:end,2:end);
p32 =cos(rs(2:end,2:end)* wr(r) -cs(2:end,2:end)*
wc(c)).* fliplr(h(2:end,2:end));
p33 = (N -rs(2:end,2:end)).* (N -cs(2:end,2:end));
p3 =sum(sum((p33.* (p31 +p32))))* 2/N^2  ;
apHW(r,c) =h(1,1) +p1 +p2 +p3;
   end
end
```

全相位处理特性主瓣能量占比、波纹幅度及过渡带宽度等指标可做具体分析。下面讨论加窗带来的传输特性改善，即在正变换前和反变换后加二维窗，如图 5-14 所示。

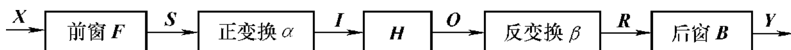


图 5-14　带前后窗的二维正交变换处理

假定系统维度为 $N\times N$，输入 $\boldsymbol{X}$ 是 $x_{i,j}$ 遍历 N^2 位置时的全部输入矩阵，$\boldsymbol{Y}$ 是对应元素 $x_{i,j}$ 位置的输出矩阵，前后窗函数定义如下：

$$\boldsymbol{F}=\begin{bmatrix} F_{0,0} & F_{0,1} & \cdots & F_{0,N-1} \\ F_{1,0} & F_{1,1} & \cdots & F_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ F_{N-1,0} & F_{N-1,1} & \cdots & F_{N-1,N-1} \end{bmatrix}_{N\times N} \quad \boldsymbol{B}=\begin{bmatrix} B_{0,0} & B_{0,1} & \cdots & B_{0,N-1} \\ B_{1,0} & B_{1,1} & \cdots & B_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ B_{N-1,0} & B_{N-1,1} & \cdots & B_{N-1,N-1} \end{bmatrix}_{N\times N} \tag{5-88}$$

则由图 5-14 可得

$$S_{p,q}=X_{p,q}^{(u,v)}\cdot F_{p,q}\quad Y_{u,v}=R_{u,v}\cdot B_{u,v}\tag{5-89}$$

由式（5-59）和式（5-60）及图 5-14 可得：

$$\begin{aligned}I_{m,n}&=\sum_{p=0}^{N-1}\sum_{q=0}^{N-1}\alpha_{m,p}\alpha_{n,q}S_{p,q}=\sum_{p=0}^{N-1}\sum_{q=0}^{N-1}\alpha_{m,p}\alpha_{n,q}F_{p,q}X_{p,q}^{(u,v)}\\ O_{m,n}&=I_{m,n}H_{m,n}=\sum_{p=0}^{N-1}\sum_{q=0}^{N-1}\alpha_{m,p}\alpha_{n,q}H_{m,n}F_{p,q}X_{p,q}^{(u,v)}\end{aligned}\tag{5-90}$$

结合式（5-61）及图 5-14，可求得输出为

$$\begin{aligned}Y_{u,v}&=B_{u,v}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\beta_{u,m}\beta_{v,n}O_{m,n}\\&=\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\beta_{u,m}\beta_{v,n}\sum_{p=0}^{N-1}\sum_{q=0}^{N-1}\alpha_{m,p}\alpha_{n,q}H_{m,n}F_{p,q}B_{u,v}X_{p,q}^{(u,v)}\\&=\sum_{p=0}^{N-1}\sum_{q=0}^{N-1}\left(\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\alpha_{m,p}\alpha_{n,q}\beta_{u,m}\beta_{v,n}H_{m,n}\mathrm{F}_{p,q}B_{u,v}\right)X_{p,q}^{(u,v)}\end{aligned}\tag{5-91}$$

所以，加双窗的全相位信号处理输出等于：

$$\begin{aligned}y&=\frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}Y_{u,v}\\&=\frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\left[\sum_{p=0}^{N-1}\sum_{q=0}^{N-1}\left(\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\alpha_{m,p}\alpha_{n,q}\beta_{u,m}\beta_{v,n}H_{m,n}F_{p,q}B_{u,v}\right)X_{p,q}^{(u,v)}\right]\end{aligned}\tag{5-92}$$

参考式（5-64）～式（5-68），可得如下全相位变换公式：

$$y=\sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1}A_{r,c}^{(d)}x_{i+r,j+c}\tag{5-93}$$

其中，全相位变换基定义如下：

$$A_{r,c}^{(d)}=\begin{cases}A^{++}=\left[\dfrac{1}{N^2}\displaystyle\sum_{u=0}^{N-1-r}\sum_{v=0}^{N-1-c}B_{u,v}T_{r+u,c+v}^{(u,v)}F_{r+u,c+v}\right]\\ \quad 1\leqslant r\leqslant N-1,\ 1\leqslant c\leqslant N-1\\ A^{+-}=\left[\dfrac{1}{N^2}\displaystyle\sum_{u=0}^{N-1-r}\sum_{v=-c}^{N-1}B_{u,v}T_{r+u,c+v}^{(u,v)}F_{r+u,c+v}\right]\\ \quad 1\leqslant r\leqslant N-1,\ -N+1\leqslant c\leqslant 0\\ A^{-+}=\left[\dfrac{1}{N^2}\displaystyle\sum_{u=-r}^{N-1}\sum_{v=0}^{N-1-c}B_{u,v}T_{r+u,c+v}^{(u,v)}F_{r+u,c+v}\right]\\ \quad -N+1\leqslant r\leqslant 0,\ 1\leqslant c\leqslant N-1\\ A^{--}=\left[\dfrac{1}{N^2}\displaystyle\sum_{u=-r}^{N-1}\sum_{v=-c}^{N-1}B_{u,v}T_{r+u,c+v}^{(u,v)}F_{r+u,c+v}\right]\\ \quad -N+1\leqslant r\leqslant 0,\ -N+1\leqslant c\leqslant 0\end{cases}\tag{5-94}$$

双窗全相位变换基由无窗变换基和以下定义的二维窗 $\boldsymbol{C}_d$ 共同决定。

$$\boldsymbol{C}^{(d)}=\begin{cases}\boldsymbol{C}^{++}=\boldsymbol{B}_{u,v}\boldsymbol{F}_{r+u,c+v}\\ \quad 1\leqslant r\leqslant N-1,\ 1\leqslant c\leqslant N-1,\ 0\leqslant u\leqslant N-1-r,\ 0\leqslant v\leqslant N-1-c\\ \boldsymbol{C}^{+-}=\boldsymbol{B}_{u,v}\boldsymbol{F}_{r+u,c+v}\\ \quad 1\leqslant r\leqslant N-1,\ -N+1\leqslant c\leqslant 0,\ 0\leqslant u\leqslant N-1-r,\ -c\leqslant v\leqslant N-1\\ \boldsymbol{C}^{-+}=\boldsymbol{B}_{u,v}\boldsymbol{F}_{r+u,c+v}\\ \quad -N+1\leqslant r\leqslant 0,\ 1\leqslant c\leqslant N-1,\ -r\leqslant u\leqslant N-1,\ 0\leqslant v\leqslant N-1-c\\ \boldsymbol{C}^{--}=\boldsymbol{B}_{u,v}\boldsymbol{F}_{r+u,c+v}\\ \quad -N+1\leqslant r\leqslant 0,\ -N+1\leqslant c\leqslant 0,\ -r\leqslant u\leqslant N-1,\ -c\leqslant v\leqslant N-1\end{cases}\tag{5-95}$$

$\boldsymbol{C}^{(d)}$ 的元素可通过图 5-15 示意图来说明。

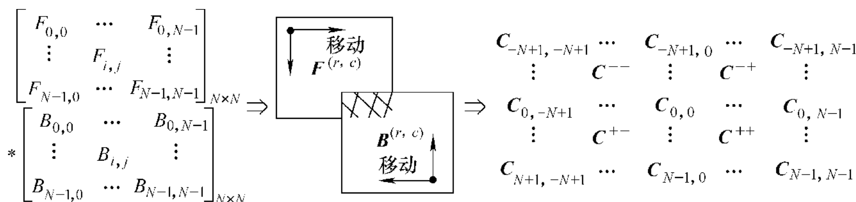


图 5-15　二维窗 $\boldsymbol{C}^{(d)}$ 形成示意图

由图 5-15 可看出，根据前后窗移动时的相对位置，分别形成窗 $\boldsymbol{C}^{(d)}$ 4 个子矩阵，其中 $\boldsymbol{C}_{r,c}{}^{(d)}$ 又是包含 $(N-|r|)(N-|c|)$ 个元素的亚矩阵，由式（5-96）定义。窗 $\boldsymbol{C}$ 形成过程类似卷积，因此称其为卷积窗。前后窗均为矩形窗时对应的卷积窗记为 $\boldsymbol{C}^{(n)}$，前或者后窗为矩形窗时的卷积窗记为 $\boldsymbol{C}^{(s)}$。

$$\boldsymbol{C}_{r,c}^{(d)}=\boldsymbol{F}_{m,n}\boldsymbol{B}_{s,t}=\begin{cases}r\leqslant m\leqslant N-1 \qquad c\leqslant n\leqslant N-1\\ 0\leqslant s\leqslant N-1-r \qquad 0\leqslant t\leqslant N-1-c\\ 1\leqslant r,\ c\leqslant N-1\\ r\leqslant m\leqslant N-1 \qquad 0\leqslant n\leqslant N-1+c\\ 0\leqslant s\leqslant N-1-r \qquad -c\leqslant t\leqslant N-1\\ 1\leqslant r\leqslant N-1,\ -N+1\leqslant c\leqslant 0\\ 0\leqslant m\leqslant N-1+r \qquad c\leqslant n\leqslant N-1\\ -r\leqslant s\leqslant N-1 \qquad 0\leqslant t\leqslant N-1-c\\ -N+1\leqslant r\leqslant 0,\ 1\leqslant c\leqslant N-1\\ 0\leqslant m\leqslant N-1-r \qquad 0\leqslant n\leqslant N-1+c\\ -r\leqslant s\leqslant N-1 \qquad -c\leqslant t\leqslant N-1\\ -N+1\leqslant r,\ c\leqslant 0\end{cases}\tag{5-96}$$

对于带双窗的二维 apDFT（简记为 $ap^{2d}DFT$），根据式（5-76）及式（5-96）可得对应的全相位变换基如下：

$$
A_{r,c}^{(d)}=\begin{cases}
A^{++}=\dfrac{h(-r,-c)}{N^2}\sum\limits_{u=0}^{N-1-r}\sum\limits_{v=0}^{N-1-c}B(u,v)F(u+r,v+c)\\
1\leqslant r\leqslant N-1,1\leqslant c\leqslant N-1\\
A^{+-}=\dfrac{h(-r,-c)}{N^2}\sum\limits_{u=0}^{N-1-r}\sum\limits_{v=-c}^{N-1}B(u,v)F(u+r,v+c)\\
1\leqslant r\leqslant N-1,-N+1\leqslant c\leqslant 0\\
A^{-+}=\dfrac{h(-r,-c)}{N^2}\sum\limits_{u=-r}^{N-1}\sum\limits_{v=0}^{N-1-c}B(u,v)F(u+r,v+c)\\
-N+1\leqslant r\leqslant 0,1\leqslant c\leqslant N-1\\
A^{--}=\dfrac{h(-r,-c)}{N^2}\sum\limits_{u=-r}^{N-1}\sum\limits_{v=-c}^{N-1}B(u,v)F(u+r,v+c)\\
-N+1\leqslant r\leqslant 0,-N+1\leqslant c\leqslant 0
\end{cases}\tag{5-97}
$$

仔细观察式（5-97），可知二重求和部分等于 $\boldsymbol{B}$ 的反转与 $\boldsymbol{F}$ 的卷积，如图 5-16所示。

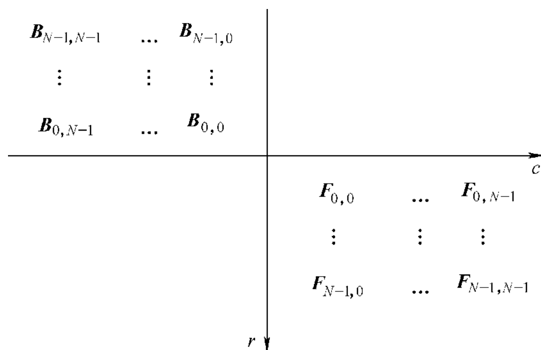


图 5-16　卷积窗 $\boldsymbol{C}_\mathrm{d}$的基窗示意图

所以式（5-97）可写成：

$$
A_{r,c}^{(d)}=\frac{h_e(-r,-c)}{N^2}\otimes\Big[B(-r,-c)*F(r,c)\Big]=h_e(-r,-c)\otimes C_{r,c}^{(d)}\tag{5-98}
$$

其中，归一化卷积窗 $C_{r,c}{}^{(d)}=B(-r,-c)*F(r,c)/N^2$。式（5-97）表明 N 阶 $ap^{2d}DFT$ 全相位基矩阵等于前后窗卷积再点乘系统冲激响应矩阵后除以 N^2。

结合式（5-82）及式（5-98），可知 $ap^{2d}DFT$ 信号处理特性函数如下：

$$
\begin{aligned}
H(j\omega_r, j\omega_c) &= \sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1} \boldsymbol{A}_{r,c}^{(d)} \mathrm{e}^{-\mathrm{j}(r\omega_r + c\omega_c)} \\
&= \boldsymbol{A}_{0,0}^{(d)} + \sum_{c=1}^{N-1}(\boldsymbol{A}_{0,c}^{(d)}\mathrm{e}^{-\mathrm{j}c\omega_c} + \boldsymbol{A}_{0,-c}^{(d)}\mathrm{e}^{\mathrm{j}c\omega_c}) + \\
&\quad \sum_{r=1}^{N-1}(\boldsymbol{A}_{r,0}^{(d)}\mathrm{e}^{-\mathrm{j}r\omega_r} + \boldsymbol{A}_{-r,0}^{(d)}\mathrm{e}^{\mathrm{j}r\omega_r}) + \\
&\quad \sum_{r=1}^{N-1}\sum_{c=1}^{N-1}\begin{bmatrix}\boldsymbol{A}_{r,-c}^{(d)}\mathrm{e}^{-\mathrm{j}(r\omega_r - c\omega_c)} + \boldsymbol{A}_{-r,c}^{(d)}\mathrm{e}^{\mathrm{j}(r\omega_r - c\omega_c)} + \\ \boldsymbol{A}_{r,c}^{(d)}\mathrm{e}^{-\mathrm{j}(r\omega_r + c\omega_c)} + \boldsymbol{A}_{-r,-c}^{(d)}\mathrm{e}^{\mathrm{j}(r\omega_r + c\omega_c)}\end{bmatrix}
\end{aligned} \tag{5-99}
$$

当全相位基满足如下条件时，特性具有线性相位。

$$
\boldsymbol{A}_{-r,-c}^{(d)} = \boldsymbol{A}_{r,c}^{(d)} \tag{5-100}
$$

特性函数又可写为

$$
\begin{aligned}
H(j\omega_r, j\omega_c) = \boldsymbol{A}_{0,0}^{(d)} + 2\sum_{c=1}^{N-1}\boldsymbol{A}_{0,c}^{(d)}\cos(c\omega_c) + 2\sum_{r=1}^{N-1}\boldsymbol{A}_{r,0}^{(d)}\cos(r\omega_r) + \\
2\sum_{r=1}^{N-1}\sum_{c=1}^{N-1}\left[\boldsymbol{A}_{r,-c}^{(d)}\cos(r\omega_r - c\omega_c) + \boldsymbol{A}_{r,c}^{(d)}\cos(r\omega_r + c\omega_c)\right]
\end{aligned} \tag{5-101}
$$

可以证明，式（5-100）是充分必要条件，使 ap^{2d}DFT 具有严格零相位。结合式（5-98），又推导出基窗和单位冲激响应需满足的条件是：

$$
\begin{aligned}
&1.\ B(N-1-r, N-1-c) = B(r,c) \text{ 和 } F(N-1-r, N-1-c) \\
&\qquad = F(r,c) \text{ 或者 } \boldsymbol{B} = \boldsymbol{F}
\end{aligned} \tag{5-102}
$$

$$
2.\ h(N-r, N-c) = h(r,c)
$$

即首先前后基窗相等或者均具有中心对称性，其次冲激响应矩阵也满足中心对称性。条件 2 的满足取决于特性矩阵 $\boldsymbol{H}$，它等价于 $\boldsymbol{H}$ 具有中心对称性。二维无窗 apDFT 处理（简记为 ap^{2n}DFT）的前后基窗均为矩形而满足条件 1，对于二维单窗 apDFT 处理（简记为 ap^{2s}DFT）则要求非矩形基窗具有中心对称性。一种可能的二维三角窗的 MATLAB 代码（画图部分省略）如下：

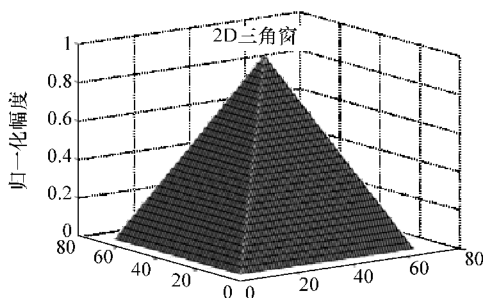
```
function out = win2D(N)
if nargin = =0
  N = 64;
end
wincore = (N - mod(N,2))/2;
for n = wincore: -1:1
    wb = ones(n,n);m = wincore - n;tlex1 = zeros(m,m);
    tlex2 = zeros(m,n);tlex3 = zeros(n,m);
    tl(:,:,1 + wincore - n) = [tlex1,tlex2;tlex3,wb];
end
tlw = sum(tl,3);
```

```
if mod(N,2)
    w2d=[tlw,fliplr(tlw(:,1:end-1));flipud(tlw(1:end-1,:)),...
        fliplr(flipud(tlw(1:end-1,1:end-1)))];
else
    w2d=[tlw,fliplr(tlw);flipud(tlw),fliplr(flipud(tlw))];
end
w2d=w2d/wincore;
```

运行结果如图 5-17 所示。以式（5-87）给出的 $\boldsymbol{H}$ 为例，ap^{2n}DFT、ap^{2s}DFT 及 ap^{2d}DFT 的处理特性如图 5-18 所示。

$$
\boldsymbol{F}=\begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1\\
1 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 1\\
1 & 2 & \ddots & \vdots & \vdots & \vdots & \vdots & \iddots & 2 & 1\\
1 & 2 & \cdots & N/2-1 & N/2-1 & N/2-1 & N/2-1 & \cdots & 2 & 1\\
1 & 2 & \cdots & N/2-1 & N/2 & N/2 & N/2-1 & \cdots & 2 & 1\\
1 & 2 & \cdots & N/2-1 & N/2 & N/2 & N/2-1 & \cdots & 2 & 1\\
1 & 2 & \cdots & N/2-1 & N/2-1 & N/2-1 & N/2-1 & \cdots & 2 & 1\\
1 & 2 & \iddots & \vdots & \vdots & \vdots & \vdots & \ddots & 2 & 1\\
1 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 1\\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{bmatrix}
$$

a) 三角窗矩阵



b) 二维三角窗

图 5-17　二维三角窗及矩阵形式

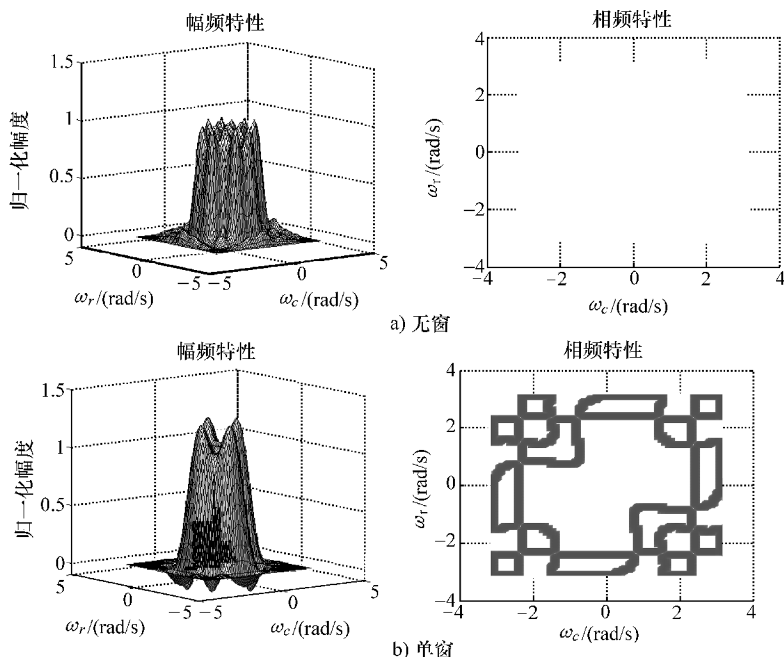
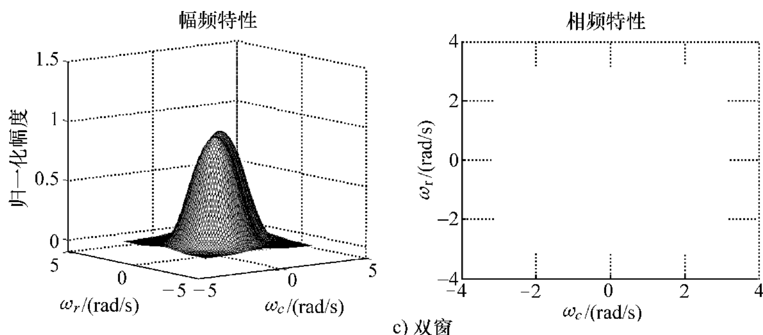


图 5-18　二维无窗、单窗和双窗 apDFT 特性



c) 双窗

图 5-18　二维无窗、单窗和双窗 apDFT 特性（续）

由图 5-18 可看出，单窗和双窗曲面的波峰和波谷数量明显小于无窗，双窗的平滑性又优于单窗；双窗和无窗的相位恒为零，单窗出现 180°的相位。由于特性具有中心对称性，图 5-19 给出了幅频特性 XZ 面投影图。

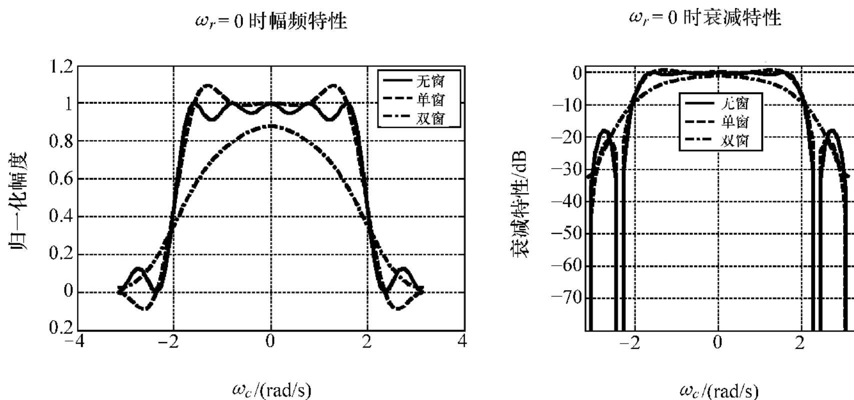


图 5-19　ap^{2n}DFT、ap^{2s}DFT 及 ap^{2d}DFT 特性 XZ 平面图

双窗的幅频特性由于采用与无窗和单窗相同的归一化方法而未达到 1，从曲线可看出其完全消除了波纹，但过渡带宽度较大；单窗的波纹相比无窗数量少，在衰减性方面没有明显区别。上面的结果与选取窗的形状有关，通过调节或者设计新窗体可进一步改善处理特性。实现上述结果的 MATLAB 代码（画图部分省略）如下：

```
function out =ap2nsdDFT(N)
if nargin = =0
  N =8;
end
h =getH(N);he =getHe(h);[w1,w2,w2nsd] =getw2(N);wn =size(w2nsd,3);
```

```
for n =1:wn
    A(:,:,n) =he.* w2nsd(:,:,n);
end
[ap2nsdHW,wr,wc] =getHW(A);
todraw(w1,w2,w2nsd,ap2nsdHW,wr,wc);
if nargout ~ =0
    out =ap2nsdHW;
end
function h =getH(N)
S =3;L =5;cs =L -2;
H =zeros(N,N);H(1,1) =1;
H(1,2:S) =1;H(1,2:end) =fliplr(H(1,2:end)) +H(1,2:end);
H(2:S,1) =1;H(2:end,1) =flipud(H(2:end,1)) +H(2:end,1);
for c =2:cs
    for r =2:L -c
        H(r,c) =1;
    end
end
H(2:end,2:end) =H(2:end,2:end) +fliplr(flipud(H(2:end,2:end)));
h =ifft2(H);
function he =getHe(h)
hro =fliplr(flipud(h));
he =[hro,hro(:,1:end -1);hro(1:end -1,:),hro(1:end -1,1:end -1)];
function [w1,w2,w2nsd] =getw2(N)
w1 =ones(N,N);w2 =win2D(N);a =blackman(N);w2 =a* a';
normn =[sum(sum(w1)),sum(sum(w2)),sum(sum(w2.* w2))];
w2nsd(:,:,1) =conv2(w1,w1);w2nsd(:,:,2) =conv2(w1,w2);
w2nsd(:,:,3) =conv2(w2,w2);
for n =1:3
    w2nsd(:,:,n) =w2nsd(:,:,n)/normn(n);
end
function [ap2nsdHW,wr,wc] =getHW(A)
wn =size(A,3);N =(size(A,1) +1)/2;
wr = -pi:pi/2^5:pi;wc = -pi:pi/2^5:pi;lenw =length(wr);
rs =repmat([0:N -1]',1,N);cs =repmat([0:N -1],N,1);
for n =1:wn
    P0 =ones(lenw,lenw)* A(N,N,n);
    P11 =2* sum(cos([1:N -1]'* wc).* repmat(A(N +1:end,N,n),1,lenw));
    P1 =repmat(P11,lenw,1);
    P21 =2* sum(cos(wr'* [1:N -1]).* repmat(A(N,N +1:end,n),lenw,1),2);
    P2 =repmat(P21,1,lenw);
```

```
    RSext = repmat(rs(2:end,2:end),lenw,lenw);
    CSext = repmat(cs(2:end,2:end),lenw,lenw);
    sz = size(rs(2:end,2:end));WRext = [];WCext = [];
    for m =1:lenw
        WRext = cat(1,WRext,ones(sz(1),sz(2))* wr(m));
        WCext = cat(2,WCext,ones(sz(1),sz(2))* wc(m));
    end
    WRexts = repmat(WRext,1,lenw);RSWR = RSext.* WRexts;
    WCexts = repmat(WCext,lenw,1);CSWC = CSext.* WCexts;
    Aextsp = repmat(A(N +1:end,N +1:end,n),lenw,lenw);
    Aextsm = repmat(A(N +1:end,N -1: -1:1,n),lenw,lenw);
    P31 = cos(RSWR + CSWC).* Aextsp + cos(RSWR - CSWC).* Aextsm;
    for r =1:lenw
        for c =1:lenw
          P3(r,c) =2* (sum(sum(P31((r -1)* sz(1) +1:r* sz(1),...
          (c -1)* sz(2) +1:c* sz(2)))));
        end
    end
    ap2nsdHW(:,:,n) = P0 + P1 + P2 + P3;
end
```

上面实验的 3 种卷积窗曲面如图 5-20 所示。

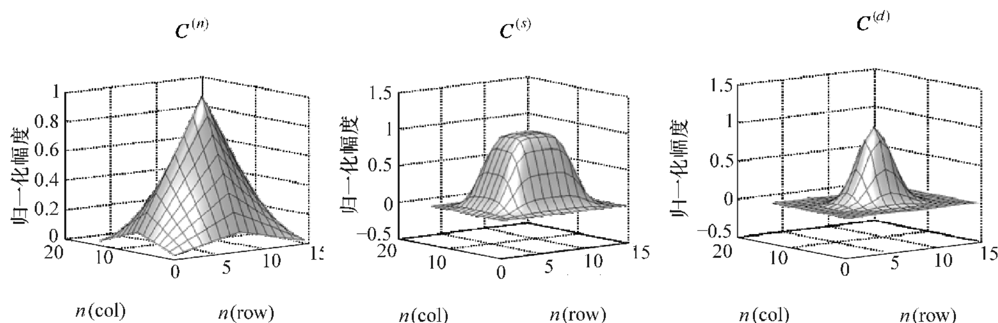


图 5-20　$C^{(n)}$、$C^{(s)}$ 及 $C^{(d)}$ 曲面图

5.5　二维 DCT/DWT 全相位

对于二维余弦变换，$\boldsymbol{\alpha} = \boldsymbol{C}$，$\boldsymbol{\beta} = \boldsymbol{C}^{\mathrm{T}}$，代入到式（5-64）并简单整理可得：

$$\boldsymbol{T}_{p,q}^{(u,v)} = \sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\boldsymbol{C}_{m,p}\,\boldsymbol{C}_{n,q}\,\boldsymbol{C}_{m,u}\,\boldsymbol{C}_{n,v}\,\boldsymbol{H}_{m,n} \tag{5-103}$$

相乘的四个余弦基元素不能进一步合并，所以余弦全相位核没有更具体的表达形

式。对于二维沃尔什变换，$\boldsymbol{\alpha}=\boldsymbol{W}_N$，$\boldsymbol{\beta}=\boldsymbol{\alpha}$，代入到式（5-64）并简单整理可得：

$$\begin{aligned}\boldsymbol{T}_{p,q}^{(u,v)} &= \sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\boldsymbol{W}_{m,p}\boldsymbol{W}_{n,q}\boldsymbol{W}_{u,m}\boldsymbol{W}_{v,n}\boldsymbol{H}_{m,n}\\ &= \sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\boldsymbol{W}_{p\otimes u,m}\boldsymbol{H}_{m,n}\boldsymbol{W}_{n,q\otimes v}\\ &= h(p\otimes u,\ q\otimes v)\end{aligned} \tag{5-104}$$

式（5-104）中符号⊗表示异或运算。对于 ap²DCT 的全相位变换核，有

$$\begin{aligned}\boldsymbol{T}_{N-1-p,N-1-q}^{(N-1-u,N-1-v)} &= \sum_{m=0}^{N-1}\sum_{n=0}^{N-1}(-1)^m\boldsymbol{C}_{m,p}(-1)^n\boldsymbol{C}_{n,q}(-1)^m\boldsymbol{C}_{m,u}(-1)^n\boldsymbol{C}_{n.v}\boldsymbol{H}_{m,n}\\ &= \boldsymbol{T}_{p,q}^{(u,v)}\\ \boldsymbol{T}_{N-1-p,q}^{(N-1-u,v)} &= \sum_{m=0}^{N-1}\sum_{n=0}^{N-1}(-1)^m\boldsymbol{C}_{m,p}\boldsymbol{C}_{n,q}(-1)^m\boldsymbol{C}_{m,u}\boldsymbol{C}_{n.v}\boldsymbol{H}_{m,n}=\boldsymbol{T}_{p,q}^{(u,v)}\\ \boldsymbol{T}_{p,N-1-q}^{(u,N-1-v)} &= \sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\boldsymbol{C}_{m,p}(-1)^n\boldsymbol{C}_{n,q}\boldsymbol{C}_{m,u}(-1)^n\boldsymbol{C}_{n.v}\boldsymbol{H}_{m,n}=\boldsymbol{T}_{p,q}^{(u,v)}\end{aligned} \tag{5-105}$$

当 N 是 2 的整数幂时，$p\otimes u=(N-1-p)\otimes(N-1-u)$，所以式（5-105）的结论对二维 apDWT 的核仍然成立。基于此结论的 ap²DCT 和 ap²DWT 的全相位变换基也具有对称性。根据式（5-94）的定义，位于第四象限的 $\boldsymbol{A}^{-}$元素 $\boldsymbol{A}_{r,c}$的中心对称是位于第一象限的 $\boldsymbol{A}^{++}$的元素 $\boldsymbol{A}_{-r,-c}$，两者的关系（假定双基窗均经过归一化处理）推导如下：

$$\begin{aligned}\boldsymbol{A}_{-r,-c}^{(d)} &= \sum_{u=0}^{N-1+r}\sum_{v=0}^{N-1+c}\boldsymbol{B}_{u,v}\boldsymbol{T}_{u-r,v-c}^{(u,v)}\boldsymbol{F}_{u-r,v-c}\\ &\overset{\substack{u'=N-1-u\\ v'=N-1-v}}{=} \sum_{u'=-r}^{N-1}\sum_{v'=-c}^{N-1}\boldsymbol{B}_{N-1-u',N-1-v'}\boldsymbol{T}_{N-1-u'-r,N-1-v'-c}^{(N-1-u',N-1-v')}\boldsymbol{F}_{N-1-u'-r,N-1-v'-c}\\ &\overset{\substack{u=u'\\ v=v'}}{=} \sum_{u=-r}^{N-1}\sum_{v=-c}^{N-1}\boldsymbol{B}_{N-1-u,N-1-v}\boldsymbol{T}_{r+u,c+v}^{(u,v)}\boldsymbol{F}_{N-1-(r+u),N-1-(c+v)}\\ &= \sum_{u=-r}^{N-1}\sum_{v=-c}^{N-1}\boldsymbol{B}_{u,v}\boldsymbol{T}_{r+u,c+v}^{(u,v)}\boldsymbol{F}_{(r+u),(c+v)} \qquad \text{如果 }\boldsymbol{B}\text{ 和 }\boldsymbol{F}\text{ 中心对称}\\ &= \boldsymbol{A}_{r,c}^{(d)}\end{aligned} \tag{5-106}$$

式（5-106）成立的条件只是要求基窗具有中心对称性，即对于任意传输矩阵 $\boldsymbol{H}$，全相位处理的特性都具有严格零相位。同样可证明，当前后基窗分别具有水平和垂直对称性时，全相位基又满足如下：

$$\boldsymbol{A}_{r,-c}^{(d)}=\boldsymbol{A}_{r,c}^{(d)} \qquad \boldsymbol{A}_{-r,c}^{(d)}=\boldsymbol{A}_{r,c}^{(d)} \tag{5-107}$$

对于第四象限的 $\boldsymbol{A}^{-}$元素 $\boldsymbol{A}_{r,c}$的转置，可推导出如下结论：

$$\begin{aligned}\boldsymbol{A}_{c,r}^{(d)} &= \sum_{u=-c}^{N-1}\sum_{v=-r}^{N-1}\boldsymbol{B}_{u,v}\boldsymbol{T}_{c+u,r+v}^{(u,v)}\boldsymbol{F}_{c+u,r+v} \\ &= \sum_{u=-r}^{N-1}\sum_{v=-c}^{N-1}\boldsymbol{B}_{v,u}\boldsymbol{T}_{c+v,r+u}^{(v,u)}\boldsymbol{F}_{c+v,r+u}\end{aligned} \tag{5-108}$$

当前后基窗满足主对称性，且全相位核满足如下条件时，$\boldsymbol{A}_{r,c}$也具有了主对称性。

$$\boldsymbol{T}_{c+v,r+u}^{(v,u)}=\boldsymbol{T}_{r+u,c+v}^{(u,v)} \tag{5-109}$$

结合式（5-103），若式（5-109）成立，则有

$$\begin{aligned}\boldsymbol{T}_{c+v,r+u}^{(v,u)} &= \sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\boldsymbol{C}_{m,c+v}\boldsymbol{C}_{n,r+u}\boldsymbol{C}_{m,v}\boldsymbol{C}_{n.u}\boldsymbol{H}_{m,n} \\ &= \sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\boldsymbol{C}_{n,c+v}\boldsymbol{C}_{m,r+u}\boldsymbol{C}_{n,v}\boldsymbol{C}_{m.u}\boldsymbol{H}_{n,m} \\ &= \sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\boldsymbol{C}_{m,r+u}\boldsymbol{C}_{n,c+v}\boldsymbol{C}_{m.u}\boldsymbol{C}_{n,v}\boldsymbol{H}_{n,m} = \boldsymbol{T}_{c+u,r+v}^{(u,v)} \\ &= \sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\boldsymbol{C}_{m,r+u}\boldsymbol{C}_{n,c+v}\boldsymbol{C}_{m.u}\boldsymbol{C}_{n,v}\boldsymbol{H}_{m,n} \\ &\Rightarrow \boldsymbol{H}_{m,n} = \boldsymbol{H}_{n,m}\end{aligned} \tag{5-110}$$

对于二维 apDWT，则有

$$\begin{aligned}\boldsymbol{T}_{q,p}^{(v,u)} &= \boldsymbol{T}_{p,q}^{(u,v)} \Rightarrow h(q\otimes v,\ p\otimes u) \\ &= h(p\otimes u,\ q\otimes v) \Rightarrow \boldsymbol{H}_{m,n}=\boldsymbol{H}_{n,m}\end{aligned} \tag{5-111}$$

综合以上，对于 $\mathrm{ap}^{2\mathrm{d}}$DCT 和 $\mathrm{ap}^{2\mathrm{d}}$DWT，我们有如下结论：

1）当前基窗 $\boldsymbol{F}$ 和后基窗 $\boldsymbol{B}$ 具有中心对称性时，全相位基中心对称；

2）当前后基窗水平/垂直对称时，全相位基水平/垂直对称；

3）当前后基窗和传输矩阵 $\boldsymbol{H}$ 主对称时，全相位基满足主对称性；

在对二维信号进行全相位处理时，基窗的选择和 $\boldsymbol{H}$ 的设定是必须考虑的问题。如果认为其他像素与中心像素的相关性与距离成反比且各向同性，则基窗应定义为中心各向对称，即同时满足水平、垂直、中心和主对称。根据信号相关性分布方向来自适应设置基窗以达到一定条件下的最优全相位信号处理效果是值得研究的内容。对于 $\boldsymbol{H}$ 的定义，则与处理的目标任务相关，即由在各个方向上不同列率信号分量的取舍来决定。如果行列上情形相同，则 $\boldsymbol{H}$ 具有主对称性。下面的讨论中，除非特别指出，前后基窗相等且具有中心各向对称性，$\boldsymbol{H}$ 也满足主对称性。假设 $\boldsymbol{H}$ 定义如下：

$$
\boldsymbol{H}=\begin{bmatrix}1&1&1&1&0&0&0&0\\1&1&1&0&0&0&0&0\\1&1&0&0&0&0&0&0\\1&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\end{bmatrix} \tag{5-112}
$$

以图 5-20 给出的 3 种卷基窗分别形成的二维 apDCT 传输特性如图 5-21 所示。

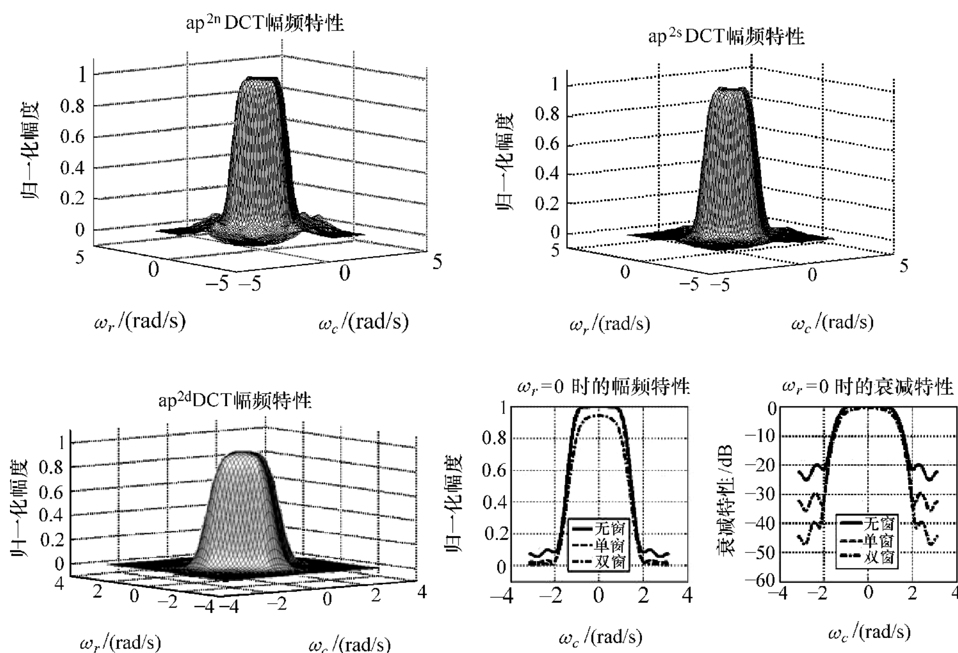


图 5-21　ap^{2nsd}DCT 特性曲面及曲线图

从图中可看出 apDCT 的平滑性好于 apDFT，ap^{2d}DCT 的衰减特性比 ap^{2n}DCT 低 20dB，但由于采用与 ap^{2n}DCT 和 ap^{2s}DCT 相同的归一化条件，峰值幅度低于 1 且过渡带宽度增大。相比而言，ap^{2s}DCT 在衰减特性和过渡带宽度都表现出良好特性。生成图 5-21 所述功能的 MATLAB 代码（只列出与 ap^{2nsd}DFT 不同的部分）如下：

```
function out = ap2nsdDCT(N)
if nargin = =0
  N =8;
```

```
end
H=getH(N);w=getw2(N);A=getA(N,H,w);
[ap2nsdHW,wr,wc]=getHW(A);
todraw(ap2nsdHW,wr,wc);
if nargout ~=0
    out=ap2nsdHW;
end

function H=getH(N)
nn=4;H=zeros(N,N);
for r=1:nn
    for c=1:nn+1-r
        H(r,c)=1;
    end
end
function A=getA(N,H,w)
A=zeros(2*N-1,2*N-1,3);
path(path,'d:/work/book1/chapter 4');
C=cosFun(N);
for nw=1:3
    F=w(:,:,nw);B=w(:,:,nw+3);
    for r=-N+1:0
        for c=r:0
            a=0;
            for u=-r:N-1
                for v=-c:N-1
                    t1=repmat(C([1:N],r+u+1).*C([1:N],u+1),1,N);
                    t2=repmat(C([1:N],c+v+1).*C([1:N],v+1),1,N);
                    T=sum(sum(t1.*t2'.*H));
                    a=a+F(u+1,v+1)*T*B(r+u+1,c+v+1);
                end
            end
            aa(r+N,c+N)=a;
        end
    end
    A1=triu(aa,1);A2=A1'+A1+diag(diag(aa));
    A(:,:,nw)=[A2,A2(1:N,N-1:-1:1);...
        A2(N-1:-1:1,1:N),A2(N-1:-1:1,N-1:-1:1)];
end
function w=getw2(N)
w(:,:,1)=ones(N,N)/N;
```

```
w(:,:,2)=w(:,:,1);w(:,:,4)=w(:,:,1);
w(:,:,3)=win2D(N);
w(:,:,5)=w(:,:,3)/sum(sum(win2D(N)))* N;
w(:,:,6)=w(:,:,3);w36c=conv2(w(:,:,3),w(:,:,6));
w(:,:,6)=w(:,:,6)/w36c(N,N);
```

与上相同条件形成的二维 ap^{2nsd}DWT 传输特性如图 5-22 所示。

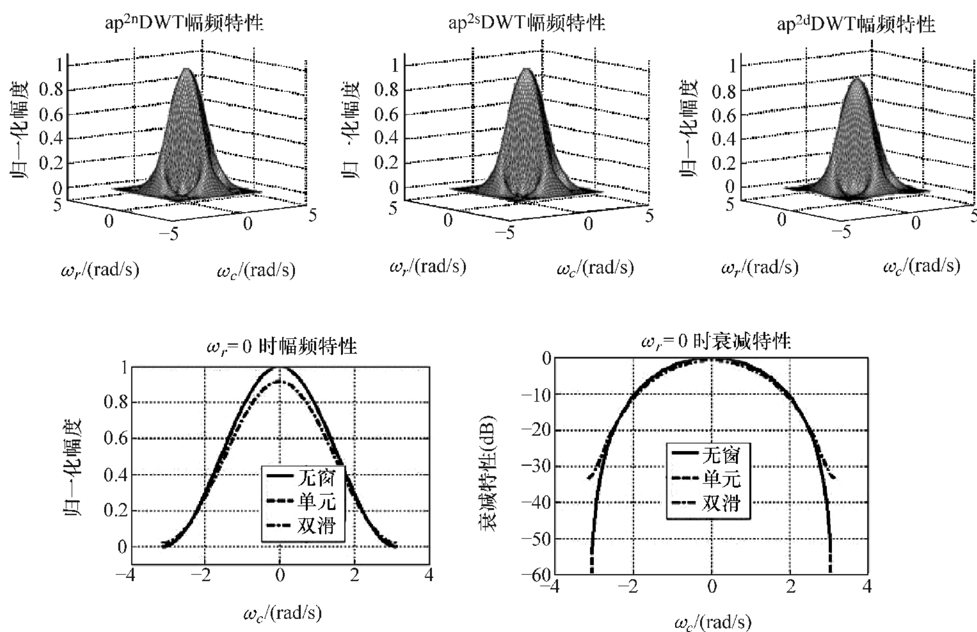


图 5-22　ap^{2nsd}DWT 特性曲面及曲线图

生成图 5-22 结果的 MATLAB 代码与 ap^{2nsd}DCT 只有函数 getA 不同，具体如下：

```
function A=getA(N,H,w)
A=zeros(2* N-1,2* N - 1,3);
path(path,'d:/work/book1/chapter 4');
wal=walFun(N);h=wal* H* wal/N^2;
for nw=1:3
    F=w(:,:,nw);B=w(:,:,nw+3);
    for r=-N+1:0
        for c=r:0
            a=0;
            for u=-r:N-1
                for v=-c:N-1
                    T=h(bitxor(r+u,u)+1,bitxor(c+v,v)+1);
```

```
                a = a + F(u + 1, v + 1) * T* B(r + u + 1, c + v + 1);
            end
        end
        aa(r + N, c + N) = a;
    end
end
A1 = triu(aa,1);A2 = A1' + A1 + diag(diag(aa));
A(:,:,nw) = [A2,A2(1:N,N - 1: - 1:1);...
    A2(N - 1: - 1:1,1:N),A2(N - 1: - 1:1,N - 1: - 1:1)];
end
```

上面给出了 DCT 和 DWT 变换情况下设计二维全相信号处理基本方法，虽然特性曲面物理意义不如 DFT 直接，即通过 $\boldsymbol{H}$ 来定义的传输特性不能与曲面特征建立直接对应关系，但提供了信号处理新思路，且具有严格零相位和优良平滑性等特点。

5.6 二维全相位信号处理特性

二维全相位在各种正交变换的基础上进行了全相位信号处理，一方面扩展了输入信号信息，另一方面对于不同相位信号进行不同权值的加权，从而使二维全相位信号处理在保留传统分块固有性质的同时也具备了一些特殊性质。

定理 1：N 阶 ap^2DFT 系统具有线性相位的充要条件是：$H(k)=H(N-k)$，且对于单窗系统要求非矩形窗中心对称，对于双窗系统要求 $\boldsymbol{F}=\boldsymbol{B}$ 或者前后窗均中心对称。N 阶 ap^2DCT/DWT 系统具有线性相位的充要条件是前后窗均中心对称。

证明：式（5-102）和式（5-106）对 DFT 和 DCT/DWT 的充分性给出了证明，这里只证明必要性。假设 ap^2 线性相位如下：

$$\varphi(\omega) = -\tau_r\omega_r - \tau_c\omega_c \tag{5-113}$$

把式（5-99）展开求得相位并整理如下：

$$\varphi(\omega) = -\arctan\frac{\sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1}[A_{r,c}\sin(r\omega_r + c\omega_c)]}{\sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1}[A_{r,c}\cos(r\omega_r + c\omega_c)]} = -\tau_r\omega_r - \tau_c\omega_c \tag{5-114}$$

$$\Rightarrow \sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1}\{A_{r,c}\sin[(r-\tau_r)\omega_r + (c-\tau_c)\omega_c]\} = 0$$

由式（5-114）可以看出，$\tau_r=\tau_c=0$ 时正弦部分关于（0，0）奇对称，式（5-114）恒成立的条件是 $\boldsymbol{A}_{r,c}$ 关于原点对称，从而得式（5-102）和式（5-106）结论，系统具有严格零相位。

定理 2：N 阶 ap^2 系统对传输特性 $\boldsymbol{H}$ 具有线性性，即若 $\boldsymbol{H}=\alpha_0\boldsymbol{H}_0+\alpha_1\boldsymbol{H}_1+\cdots+$

$\alpha_{N-1}\boldsymbol{H}_{N-1}$，则全相位处理特性 ap$\boldsymbol{H}$ 等于各子处理特性 ap$\boldsymbol{H}_i$的加权和即 ap$\boldsymbol{H}=\alpha_0$ ap$\boldsymbol{H}_0+\alpha_1ap\boldsymbol{H}_1+\cdots+\alpha_{N-1}ap\boldsymbol{H}_{N-1}$。

证明：式（5-99）说明 ap$\boldsymbol{H}$ 对全相位变换基 $\boldsymbol{A}$ 具有线性性，而由式（5-94）又可知 $\boldsymbol{A}$ 对全相位变换核 $\boldsymbol{T}$ 具有线性性，由式（5-91）又有

$$\begin{aligned}\boldsymbol{T} &= \sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\boldsymbol{\alpha}_{m,p}\boldsymbol{\alpha}_{n,q}\boldsymbol{\beta}_{u,m}\boldsymbol{\beta}_{v,n}\boldsymbol{H}_{m,n}\boldsymbol{F}_{p,q}\boldsymbol{B}_{u,v}\\ &= \sum_{m=0}^{N-1}\sum_{m=0}^{N-1}\boldsymbol{\alpha}_{m,p}\boldsymbol{\alpha}_{n,q}\boldsymbol{\beta}_{u,m}\boldsymbol{\beta}_{v,n}\sum_{i=0}^{N-1}\boldsymbol{H}_{m,n}^{(i)}\boldsymbol{F}_{p,q}\boldsymbol{B}_{u,v}\\ &= \sum_{i=0}^{N-1}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\boldsymbol{\alpha}_{m,p}\boldsymbol{\alpha}_{n,q}\boldsymbol{\beta}_{u,m}\boldsymbol{\beta}_{v,n}\boldsymbol{H}_{m,n}^{(i)}\boldsymbol{F}_{p,q}\boldsymbol{B}_{u,v} = \sum_{i=0}^{N-1}{}^{(i)}\boldsymbol{T}\end{aligned}\tag{5-115}$$

综上，二维全相位处理特性 ap$\boldsymbol{H}$ 对传输特性 $\boldsymbol{H}$ 具有线性性。利用此性质，可在行列方向上形成高、低通滤波器对二维信号如图像进行全相位处理从而将图像分解。以 8 阶 ap^{2}DCT 为例，行列方向频率二等分的四个子传输特性如下：

$$\boldsymbol{H}_{LL}=\begin{bmatrix}\mathbf{1} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix},\ \boldsymbol{H}_{HL}=\begin{bmatrix}\mathbf{0} & \mathbf{1}\\ \mathbf{0} & \mathbf{0}\end{bmatrix},\ \boldsymbol{H}_{LH}=\begin{bmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{1} & \mathbf{0}\end{bmatrix},\ \boldsymbol{H}_{HH}=\begin{bmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{1}\end{bmatrix}\tag{5-116}$$

式（5-116）中 **0** 和 **1** 均表示 4×4 矩阵。对应 4 个全相位处理特性 ap$\boldsymbol{H}_i$（$i=0\sim3$）如图 5-23 所示。

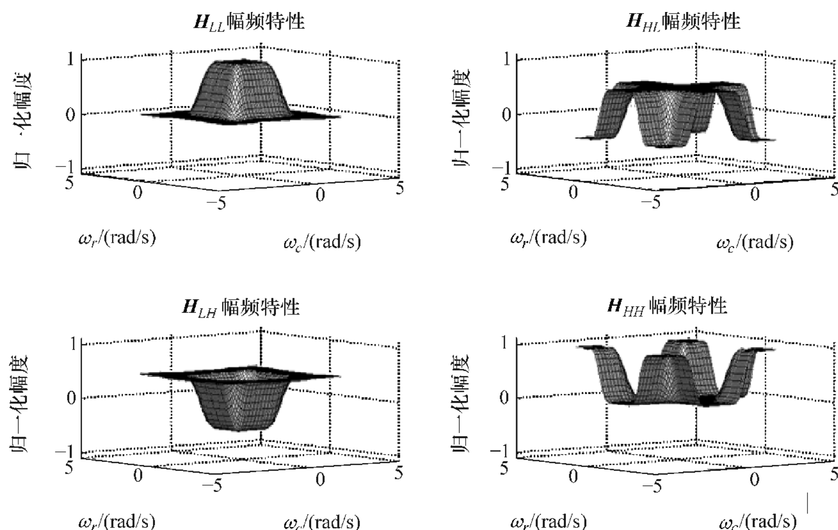


图 5-23　ap^{2}DCT 四个子带处理特性曲面

此性质可用于设计图像多分辨率分解的全相位滤波器，相比于 ap^{1}只在各列进行处理而言，ap^{2}更充分利用了行列间二维相关性。

定理 3：N 阶 ap^2 系统对基窗 $\boldsymbol{F}/\boldsymbol{B}$ 具有线性性。若 $\boldsymbol{F}=\alpha_0\boldsymbol{F}_0+\alpha_1\boldsymbol{F}_1+\cdots+$

$\alpha_{N-1}\boldsymbol{F}_{N-1}$，则全相位处理特性 ap$\boldsymbol{H}$ 等于各子处理特性 ap$\boldsymbol{H}_i$的加权和，即 ap$\boldsymbol{H}$ = $\alpha_0$ap$\boldsymbol{H}_0$ + $\alpha_1$ap$\boldsymbol{H}_1$ + ⋯ + α_{N-1}ap$\boldsymbol{H}_{N-1}$，其中 ap$\boldsymbol{H}_i$是对应 $\alpha_i\boldsymbol{F}_i$基窗的处理特性。

证明：式（5-99）说明 ap$\boldsymbol{H}$ 对全相位变换基 $\boldsymbol{A}$ 具有线性性，而由式（5-93）又有：

$$\begin{aligned}\boldsymbol{A}_{r,c} &= \sum_{u=0}^{R}\sum_{v=0}^{C}\boldsymbol{B}_{u,v}\boldsymbol{T}_{r+u,c+v}^{(u,v)}\boldsymbol{F}_{r+u,c+v} \\ &= \sum_{u=0}^{R}\sum_{v=0}^{C}\sum_{i=0}^{N-1}\alpha_i\boldsymbol{B}_{u,v}^{(i)}\boldsymbol{T}_{r+u,c+v}^{(u,v)}\boldsymbol{F}_{r+u,c+v} \\ &= \sum_{i=0}^{N-1}\alpha_i\sum_{u=0}^{R}\sum_{v=0}^{C}\boldsymbol{B}_{u,v}^{(i)}\boldsymbol{T}_{r+u,c+v}^{(u,v)}\boldsymbol{F}_{r+u,c+v} = \sum_{i=0}^{N-1}\alpha_i\boldsymbol{A}_{r,c}^{(i)}\end{aligned} \tag{5-117}$$

所以，二维全相位处理特性 ap$\boldsymbol{H}$ 对基窗 $\boldsymbol{F}/\boldsymbol{B}$ 具有线性性。可把基窗分解成若干个子基窗，从而得到信号不同加权情况下的全相位处理输出，其特性取决于子基窗的形状。以式（5-116）中低通特性 $\boldsymbol{H}_{LL}$为例，为保持全相位处理的严格零相位特性，仍使子基窗满足中心对称。对图 5-17 定义的三角窗进行分解的四个子基窗如下：

$$\boldsymbol{B}_0 = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 \\ 1 & 1 & \cdots & 1 & 1 \\ \vdots & \vdots & \boldsymbol{1} & \vdots & \vdots \\ 1 & 1 & \cdots & 1 & 1 \\ 1 & 1 & \cdots & 1 & 1 \end{bmatrix},\ \boldsymbol{B}_1 = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 1 & 0 \\ \vdots & \vdots & \boldsymbol{1} & \vdots & \vdots \\ 0 & 1 & \cdots & 1 & 0 \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix},$$
$$\boldsymbol{B}_2 = \begin{bmatrix} \ddots & & \cdots & & \iddots \\ & 1 & 1 & 1 & 1 & \\ & 1 & & & 1 & \\ \vdots & & \boldsymbol{1} & & & \vdots \\ & 1 & & & 1 & \\ & 1 & 1 & 1 & 1 & \\ \iddots & & \cdots & & \ddots \end{bmatrix},\ \boldsymbol{B}_3 = \begin{bmatrix} \ddots & 0 & 0 & \iddots \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ \iddots & 0 & 0 & \ddots \end{bmatrix} \tag{5-118}$$

式（5-118）中 $\boldsymbol{1}$ 表示 4×4 矩阵。矩阵 $\boldsymbol{B}_i$ 的外围 i 周为 0，中心为 $(8-2i)\times(8-2i)$ 的单位阵，$\boldsymbol{B}_i$也可以理解由 4×4 单位矩阵 $\boldsymbol{B}_3$经过三次膨胀而形成各子矩阵。以子基窗 $\boldsymbol{B}_i$为后窗对应的 4 个 ap$\boldsymbol{H}_i$特性和剖面图（$\omega_r=0$）如图 5-24 所示。

由图 5-25 可看出，不同子基窗对应的特性在主瓣能量占比、带内波动性和过渡带宽度等方面都发生了变化，其综合效果是对性能的某种程度改善。因此，通过调整基窗形状来优化全相位处理特性是值得研究的内容。

定理 4：N 阶 ap^{2}DFT 系统频率响应函数等于频率采样向量 $\boldsymbol{H}$ 与卷积窗 $\boldsymbol{C}$ 的离散卷积。即频率响应函数可通过频率采样序列内插实现，其内插函数为卷积窗谱函数。

证明：把式（5-98）代入式（5-99）得 ap^{2}DFT 系统频率响应如下：

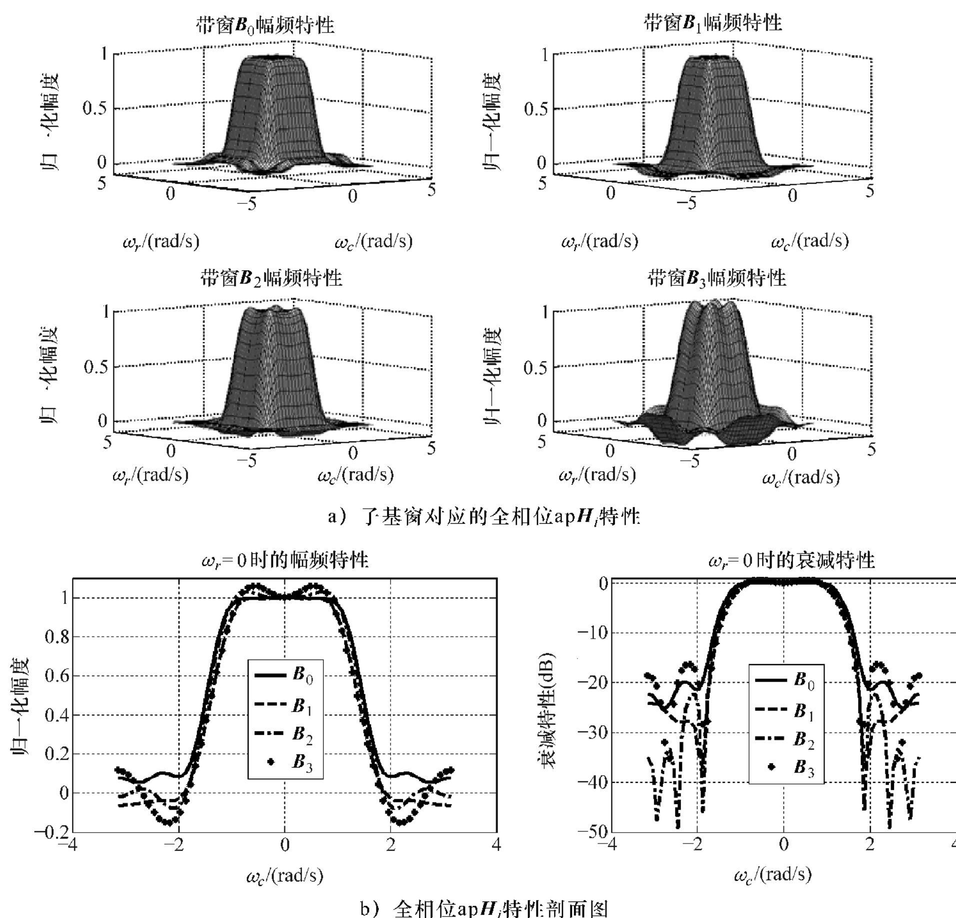


图 5-24　ap^{2}DCT 四子窗对应 ap$\boldsymbol{H}_i$特性

$$
\begin{aligned}
\mathrm{ap}H(\mathrm{j}\omega_r,\ \mathrm{j}\omega_c) &= \sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1}\boldsymbol{C}_{r,c}h(-r,\ -c)\mathrm{e}^{-\mathrm{j}(r\omega_r+c\omega_c)} \\
&= \sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1}\boldsymbol{C}_{r,c}\left[\frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\boldsymbol{H}_{u,v}\mathrm{e}^{-\mathrm{j}(2\pi\cdot ur/N+2\pi\cdot vc/N)}\right]\mathrm{e}^{-\mathrm{j}(r\omega_r+c\omega_c)} \\
&= \frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\boldsymbol{H}_{u,v}\left[\sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1}\boldsymbol{C}_{r,c}\mathrm{e}^{-\mathrm{j}(\omega_r+2\pi\cdot u/N)r-\mathrm{j}(\omega_c+2\pi\cdot v/N)c}\right] \\
&= \frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\boldsymbol{H}_{u,v}\varPhi[\mathrm{j}(\omega_r+2u\pi/N),\ \mathrm{j}(\omega_c+2v\pi/N)]
\end{aligned}
\tag{5-119}
$$

式中，$\varPhi$（ω_r，ω_c）是卷积窗 $\boldsymbol{C}$ 的谱。由式（5-98）可知，$\varPhi$（ω_r，ω_c）等于前基窗谱和后基窗共轭谱乘积，即

$$
\varPhi(\mathrm{j}\omega_r,\ \mathrm{j}\omega_c) = F(\mathrm{j}\omega_r,\ \mathrm{j}\omega_c)\cdot B^*(\mathrm{j}\omega_r,\ \mathrm{j}\omega_c) \tag{5-120}
$$

对于 ap^{2n}DFT，前后窗谱如下：

$$F(\mathrm{j}\omega_r,\ \mathrm{j}\omega_c)=B(\mathrm{j}\omega_r,\ \mathrm{j}\omega_c)=\mathrm{e}^{-\mathrm{j}(N-1)(\omega_r+\omega_c)/2}\frac{\sin(N\omega_r/2)}{N\sin(\omega_r/2)}\frac{\sin(N\omega_c/2)}{N\sin(\omega_c/2)} \tag{5-121}$$

所以，ap^2DFT 系统频率响应函数等于

$$\begin{aligned}
\text{ap}H(\mathrm{j}\omega_r,\mathrm{j}\omega_c) &= \frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\boldsymbol{H}_{u,v}\Phi[\mathrm{j}(\omega_r+2u\pi/N),\ \mathrm{j}(\omega_c+2v\pi/N)]\\
&= \frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\boldsymbol{H}_{u,v}\left[\frac{\sin(N\omega_{r'}/2)}{N\sin(\omega_{r'}/2)}\right]^2\left[\frac{\sin(N\omega_{c'}/2)}{N\sin(\omega_{c'}/2)}\right]^2,\ \omega_{r'(c')}\\
&= \omega_{r(c)}+2u(v)\pi/N\\
&= \frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\boldsymbol{H}_{u,v}\left[\frac{\sin(N\omega_{r'}/2-N\pi)}{N\sin(\omega_{r'}/2-\pi)}\right]^2\\
&\quad\left[\frac{\sin(N\omega_{c'}/2-N\pi)}{N\sin(\omega_{c'}/2-\pi)}\right]^2,\ \omega_{r(c)}=\omega_{r'(c')}-2\pi(N-u(v))/N\\
&= \frac{1}{N^2}\sum_{u'=1}^{N}\sum_{v'=1}^{N}\boldsymbol{H}_{N-u',N-v'}\left[\frac{\sin(N\omega_r/2)}{N\sin(\omega_r/2)}\right]^2\left[\frac{\sin(N\omega_c/2)}{N\sin(\omega_c/2)}\right]^2,\ u'(v')\\
&= N-u(v)
\end{aligned} \tag{5-122}$$

从式（5-122）可以看出，插值函数 $\Phi(\omega_r,\ \omega_c)$ 只有在 $r=u'$ 和 $c=v'$ 时有非零值 1，所以，线性 ap^2DFT 系统频率响应函数在（$2r\pi/N$，$2c\pi/N$）的取样值等于：

$$\text{ap}\,H(\mathrm{j}\omega_r,\mathrm{j}\omega_c)\big|_{r=u',c=v'}=\frac{1}{N^2}\sum_{u'=1}^{N}\sum_{v'=1}^{N}\boldsymbol{H}_{N-r,N-c}=\frac{N^2}{N^2}\boldsymbol{H}_{r,c}=\boldsymbol{H}_{r,c} \tag{5-123}$$

式（5-123）表示 ap^2DFT 系统频率响应函数经过 $\boldsymbol{H}$ 每个采样点。为使式（5-119）恒成立，卷积窗也需要满足一定的条件。

定理 5：N 阶 ap^{2ns}DFT 系统频率响应函数满足 $\text{ap}H(\omega_r,\omega_c)\,|_{\omega_r=2r\pi/N,\ \omega_c=2c\pi/N,}=\boldsymbol{H}_{r,c}$ 的充要条件是卷积窗 $\boldsymbol{C}$ 中心点等于 1；apDW 则满足 $\text{ap}H(\omega_r,\omega_c)\,|_{\omega_r=2r\pi/N,\ \omega_c=2c\pi/N,}<\boldsymbol{H}_{r,c}$。

证明：由式（5-119）可得如下：

$$\begin{aligned}
\text{ap}H\left(\mathrm{j}\frac{2u\pi}{N},\ \mathrm{j}\frac{2v\pi}{N}\right) &= \sum_{r=-N+1}^{N-1}\sum_{c=-N+1}^{N-1}\boldsymbol{C}_{r,c}h(-r,\ -c)\mathrm{e}^{-\mathrm{j}\left(r\frac{2u\pi}{N}+c\frac{2v\pi}{N}\right)}\\
&= \sum_{r=-N+1}^{N-1}\mathrm{e}^{-\mathrm{j}r\frac{2u\pi}{N}}\left[\boldsymbol{C}_{r,0}h(-r,0)+\sum_{c=1}^{N-1}(\boldsymbol{C}_{r,c}+\boldsymbol{C}_{r,c-N})h(-r,\ -c)\mathrm{e}^{-\mathrm{j}c\frac{2v\pi}{N}}\right]\\
&= \boldsymbol{C}_{0,0}h(0,0)+\sum_{c=1}^{N-1}(\boldsymbol{C}_{0,c}+\boldsymbol{C}_{0,c-N})h(0,\ -c)\mathrm{e}^{-\mathrm{j}c\frac{2v\pi}{N}}\\
&\quad+\sum_{r=1}^{N-1}(\boldsymbol{C}_{r,0}+\boldsymbol{C}_{r-N,0})h(-r,\ 0)\mathrm{e}^{-\mathrm{j}r\frac{2u\pi}{N}}\\
&\quad+\sum_{r=1}^{N-1}\sum_{c=1}^{N-1}(\boldsymbol{C}_{r,c}+\boldsymbol{C}_{r,c-N}+\boldsymbol{C}_{r-N,c}+\boldsymbol{C}_{r-N,c-N})h(-r,\ -c)\mathrm{e}^{-\mathrm{j}r\frac{2u\pi}{N}-\mathrm{j}c\frac{2v\pi}{N}}
\end{aligned} \tag{5-124}$$

对于 ap^{2ns}DFT，结合图 5-16 和图 5-17 可知卷积窗与前基窗 $\boldsymbol{f}$ 的关系如下：

$$\boldsymbol{C}_{0,0} = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)$$

$$\boldsymbol{C}_{0,c} = \sum_{x=0}^{N-1}\sum_{y=c}^{N-1} f(x,y),\ \boldsymbol{C}_{0,c-N} = \sum_{x=0}^{N-1}\sum_{y=0}^{c-1} f(x,y) \Rightarrow \boldsymbol{C}_{0,c} + \boldsymbol{C}_{0,c-N} = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y) = \boldsymbol{C}_{0,0}$$

$$\boldsymbol{C}_{r,0} = \sum_{x=r}^{N-1}\sum_{y=0}^{N-1} f(x,y),\ \boldsymbol{C}_{r-N,0} = \sum_{x=0}^{r-1}\sum_{y=0}^{N-1} f(x,y) \Rightarrow \boldsymbol{C}_{r,0} + \boldsymbol{C}_{r-N,0} = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y) = \boldsymbol{C}_{0,0}$$

$$\boldsymbol{C}_{r,c} = \sum_{x=r}^{N-1}\sum_{y=c}^{N-1} f(x,y),\ \boldsymbol{C}_{r-N,c} = \sum_{x=0}^{r-1}\sum_{y=c}^{N-1} f(x,y) \Rightarrow \boldsymbol{C}_{r,c} + \boldsymbol{C}_{r-N,c} = \sum_{x=0}^{N-1}\sum_{y=c}^{N-1} f(x,y)$$

$$\boldsymbol{C}_{r,c-N} = \sum_{x=r}^{N-1}\sum_{y=0}^{c-1} f(x,y),\ \boldsymbol{C}_{r-N,c-N} = \sum_{x=0}^{r-1}\sum_{y=0}^{c-1} f(x,y) \Rightarrow \boldsymbol{C}_{r,c-N} + \boldsymbol{C}_{r-N,c-N} = \sum_{x=0}^{N-1}\sum_{y=0}^{c-1} f(x,y)$$

$$\Rightarrow \boldsymbol{C}_{r,c} + \boldsymbol{C}_{r-N,c} + \boldsymbol{C}_{r,c-N} + \boldsymbol{C}_{r-N,c-N} = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y) = \boldsymbol{C}_{0,0} \tag{5-125}$$

将式（5-125）的结果代入式（5-124），并经简单整理得到：

$$\begin{aligned} \mathrm{ap}H\left(\mathrm{j}\frac{2u\pi}{N}, \mathrm{j}\frac{2v\pi}{N}\right) &= \boldsymbol{C}_{0,0} \sum_{r=0}^{N-1}\sum_{c=0}^{N-1} h(-r,-c)\mathrm{e}^{-\mathrm{j}r\frac{2u\pi}{N}-\mathrm{j}c\frac{2v\pi}{N}} \\ &= \boldsymbol{C}_{0,0} H(N-u, N-v) \Leftrightarrow \boldsymbol{C}_{0,0} = 1 \end{aligned} \tag{5-126}$$

但是，对于 ap^{2d}DFT 来说，由于前后基窗均是非矩形平面，不再具有式（5-125）的结论。按照与上面的方法得到卷积窗 $\boldsymbol{C}$ 与前基窗 $\boldsymbol{f}$ 和后基窗 $\boldsymbol{b}$ 关系如下：

$$\begin{aligned} &\boldsymbol{C}_{0,0} = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)b(x,y) \\ &\boldsymbol{C}_{0,c} = \sum_{x=0}^{N-1}\sum_{y=c}^{N-1} f(x,y)b(x,y-c),\ \boldsymbol{C}_{0,c-N} = \sum_{x=0}^{N-1}\sum_{y=0}^{c-1} f(x,y)b(x,N+y-c) \\ &\quad = \sum_{x=0}^{N-1}\sum_{y=0}^{c-1} f(x,y)b(x,y-c) \\ &\quad \Rightarrow \boldsymbol{C}_{0,c} + \boldsymbol{C}_{0,c-N} = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)b(x,y-c) \\ &\boldsymbol{C}_{r,0} + \boldsymbol{C}_{r-N,0} = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)b(x-r,y) \\ &\boldsymbol{C}_{r,c} + \boldsymbol{C}_{r-N,c} + \boldsymbol{C}_{r,c-N} + \boldsymbol{C}_{r-N,c-N} = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)b(x-r,y-c) \end{aligned} \tag{5-127}$$

对于线性相位 ap^{2d}DFT 即 $\boldsymbol{f}=\boldsymbol{b}$，由式（5-127）易知：

$$\boldsymbol{C}_{0,0} \geqslant \boldsymbol{C}_{0,c} + \boldsymbol{C}_{0,c-N}(\text{或}\boldsymbol{C}_{r,0} + \boldsymbol{C}_{r-N,0}) \geqslant \boldsymbol{C}_{r,c} + \boldsymbol{C}_{r-N,c} + \boldsymbol{C}_{r,c-N} + \boldsymbol{C}_{r-N,c-N} \tag{5-128}$$

显然，如果此时用 $\boldsymbol{C}_{0,0}$对式（5-125）进行归一化，由式（5-128）的结论可知全相位特性在频率采样点的值小于频率采样，即 $ap\mathrm{H}(e^{j\omega})\big|_{\omega=2n\pi/N}<\mathrm{H}(n)$。以图 5-17 定义的三角窗为例，画出$(\boldsymbol{C}_{r,c}+\boldsymbol{C}_{r-N,c}+\boldsymbol{C}_{r,c-N}+\boldsymbol{C}_{r-N,c-N})/\boldsymbol{C}_{0,0}$，如图 5-25 所示。

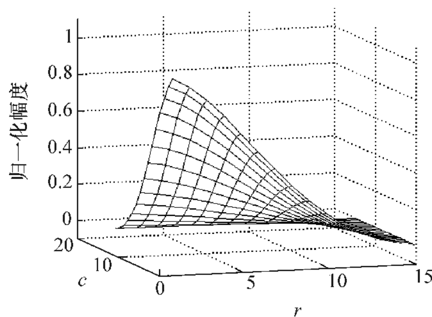


图 5-25　三角窗归一化自相关曲面

实现图 5-25 的 MATLAB 代码（省略画图部分）如下：

```
functiondrawEn2(N)
% this function is to draw En and its distance to 1
if nargin = =0
   N =16;
end
F =win2D(N);
C00 =sum(sum(F.* F));
for x =1:N -1
   CX(x) =sum(sum(F(:,x +1:N).* F(:,1:N -x)));
end
for y =1:N -1
   CY(y) =sum(sum(F(y +1:N,:).* F(1:N -y,:)));
end
for x =1:N -1
     for y =1:N -1
         CXY(x,y) =sum(sum(F(y +1:N,x +1:N).* F(1:N -y,1:N -x)));
   end
end
```

定理 6：二维数据经过 W-O 处理后的自相关性降低，卷积双窗时降幅最大。

证明：二维信号 W-O 处理过程如图 5-26 所示。

无限长宽序列 $\boldsymbol{x}$ 的自相关函数可取$(2N-1)\times(2N-1)$个数据来进行衡量，并假设相关性平稳即与时间起始点无关，可表示如下：

$$r_{xx}(m,n) = \frac{1}{(2N-1)^2}\sum_{i=-N+1}^{N-1}\sum_{j=-N+1}^{N-1} x(i,j)x^*(i+m,j+n) \tag{5-129}$$

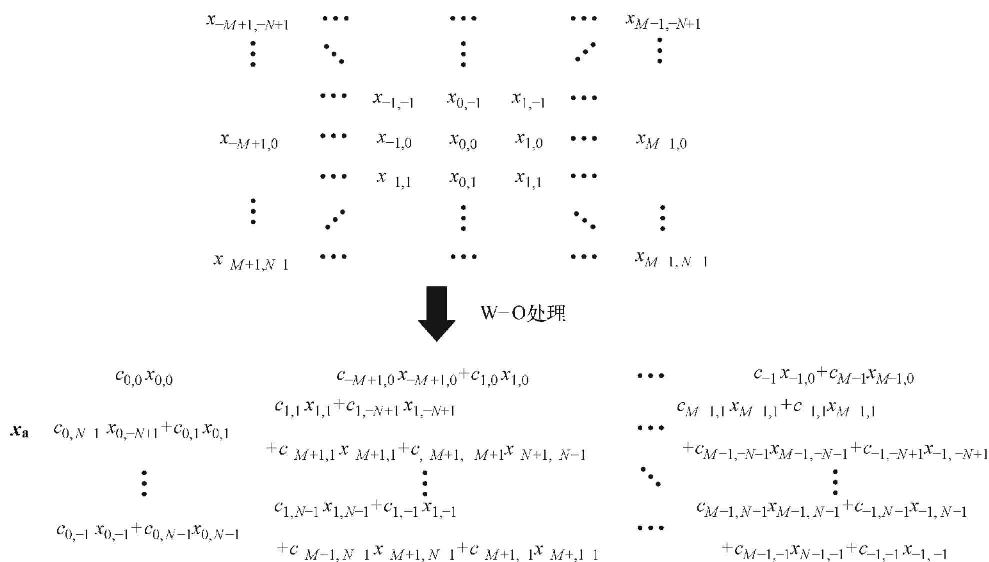


图 5-26　二维 W-O 处理示意图

全相位变换后信号 $\boldsymbol{x}_a$ 的自相关等于：

$$\begin{aligned} r_{x_a x_a}(m,n) &= \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} x_a(i,j)x_a^*(i+m,j+n) \\ &= \frac{1}{N^2}\left[c_{0,0}^2 x_{0,0}x_{0,0}^* + \sum_{j=1}^{N-1}(c_{0,j}x_{0,j} + c_{0,-N+j}x_{0,-N+j})(c_{0,j}x_{m,n+j}^* + c_{0,-N+j}x_{m,-N+n+j}^*)\right] \\ &+ \frac{1}{N^2}\left[\sum_{i=1}^{N-1}(c_{i,0}x_{i,0} + c_{-M+i,0}x_{-M+i,0})(c_{i,0}x_{m+i,n}^* + c_{-M+i,0}x_{-M+m+i,n}^*)\right] \\ &+ \frac{1}{N^2}\sum_{i=1}^{N-1}\sum_{j=1}^{N-1}(c_{i,j}x_{i,j} + c_{-M+i,j}x_{-M+i,j} + c_{i,-N+j}x_{i,-N+j} + c_{-M+i,-N+j}x_{-M+i,-N+j}) \cdot \\ &(c_{i,j}x_{i+m,j+n}^* + c_{-M+i,j}x_{-M+i+m,j+n}^* + c_{i,-N+j}x_{i+m,-N+n+j}^* + c_{-M+i,-N+j}x_{-M+m+i,-N+n+j}^*) \end{aligned} \tag{5-130}$$

设 $\boldsymbol{x}$ 为确定复指数信号矩阵，用最大幅值进行归一化后的元素 $x_{i,j} = AW_i^{\alpha i+\varphi} W_j^{\beta j+\theta}$（其中 $W_i = \mathrm{e}^{-\mathrm{j}2\pi/M}$，$W_j = \mathrm{e}^{-\mathrm{j}2\pi/N}$，$\alpha$、$\beta$、$\varphi$、$\theta$ 为实变量）代入式（5-129）并令 $m=n=0$，可得，$\boldsymbol{x}$ 最大自相关为

$$r_{xx}(0,0) = \frac{1}{(2N-1)^2}\sum_{i=-N+1}^{N-1}\sum_{j=-N+1}^{N-1} AW_i^{\alpha i+\varphi}W_j^{\beta j+\theta}AW_i^{-\alpha i-\varphi}W_j^{-\beta j-\theta} = A^2 \tag{5-131}$$

将 $x_{i,j}$ 表达式代入式（5-129）并令 $m=n=0$ 可得 $\boldsymbol{x}_a$ 最大自相关为

$$
\begin{aligned}
r_{x_a x_a}(0,0) &= \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} x_a(i,j)x_a^*(i+m,j+n) \\
&= \frac{A^2}{N^2}\begin{Bmatrix} c_{0,0}^2 + \sum_{j=1}^{N-1}\left[c_{0,j}^2 + 2c_{0,j}c_{0.-N+j}\cos(2\beta\pi) + c_{0,-N+j}^2\right] \\ + \sum_{i=1}^{N-1}\left[c_{i,0}^2 + 2c_{i,0}c_{-M+i.0}\cos(2\alpha\pi) + c_{-M+i,0}^2\right] \end{Bmatrix} \\
&+ \frac{A^2}{N^2}\sum_{i=1}^{N-1}\sum_{j=1}^{N-1}\left[c_{i,j}^2 + c_{-M+i,j}^2 + c_{i,-N+j}^2 + c_{-M+i,-N+j}^2\right] \\
&+ \frac{A^2}{N^2}\sum_{i=1}^{N-1}\sum_{j=1}^{N-1}\left[2c_{i,j}c_{-M+i,j}\cos(2\alpha\pi) + 2c_{i,j}c_{i,-N+j}\cos(2\beta\pi)\right. \\
&\left. + 2c_{i,j}c_{-M+i,-N+j}\cos(2\alpha\pi + 2\beta\pi)\right] \\
&+ \frac{A^2}{N^2}\sum_{i=1}^{N-1}\sum_{j=1}^{N-1}\left[2c_{-M+i,j}c_{i,-N+j}\cos(2\alpha\pi - 2\beta\pi) + 2c_{-M+i,j}c_{-M+i,-N+j}\cos(2\beta\pi)\right. \\
&\left. + 2c_{i,-N+j}c_{-M+i,-N+j}\cos(2\alpha\pi)\right]
\end{aligned}
\tag{5-132}
$$

将式（5-131）的结论代入到式（5-132）中，并利用三角函数小于 1 的性质可得：

$$
\frac{r_{x_a x_a}(0,0)}{r_{xx}(0,0)} \leqslant \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\left(c_{i,j} + c_{-M+i,j} + c_{i,-N+j} + c_{-M+i,-N+j}\right)^2 c_{i,-N} = c_{-M,j} = c_{-M,-N} = 0 \tag{5-133}
$$

由式（5-125）和式（5-126）的结论知，对于 $\mathrm{ap}^{2\mathrm{ns}}$ 的归一化卷积窗 $\boldsymbol{C}$ 满足：

$$c_{i,j} + c_{-M+i,j} + c_{i,-N+j} + c_{-M+i,-N+j} = 1 \tag{5-134}$$

由式（5-128）的结论可知，对于 $\mathrm{ap}^{2\mathrm{d}}$ 的归一化卷积窗 $\boldsymbol{C}$ 满足：

$$c_{i,j} + c_{-M+i,j} + c_{i,-N+j} + c_{-M+i,-N+j} < 1 \tag{5-135}$$

因此，对于 $\mathrm{ap}^{2\mathrm{nsd}}$ 处理后数据的相关性均减小，即满足：

$$r_{x_a x_a}(0,0) \leqslant r_{xx}(0,0) \tag{5-136}$$

由于双窗时的相关性进行了第二次降低，因此 $\mathrm{ap}^{2\mathrm{d}}$ 处理具有最大降幅。

假设输入的复信号为实部和虚部均是均值为 0，且方差为 1 的高斯白噪声，采用 N^2 阶进行 W-O 处理后的信号对比原信号自相关结果如图 5-27 所示。

实现图 5-27 所示结果的 MATLAB 代码如下：

```
function wo2Rxx(N)

clc,clf;
if nargin = =0
```

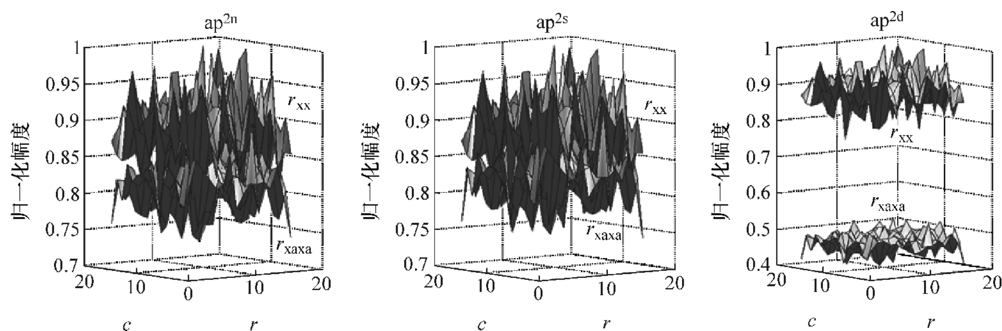


图 5-27　二维 W-O 处理后相关性对比

```
    N =16;slen =N^2 +2* N -1;fontsize =16;
end
% generateconvulation window
fwin(:,:,1) =boxcar(N)* (boxcar(N))';fwin(:,:,2) =triang(N)* triang(N)';
bwin(:,:,1) =boxcar(N)* (boxcar(N))';bwin(:,:2) =win2D(N);
for i =1:2
    for j =1:2
        C(:,:,i +j -1) =conv2(fwin(:,:,i),bwin(:,:,j));
        C(:,:,i +j -1) =C(:,:,i +j -1)/C(N,N,i +j -1);
    end
end

sigin =rand(slen,slen) +j* rand(slen,slen);
% to get the self relation
[rxx,ryy] =getrxy(N,sigin,C);

tname ={ 'ap^2^n','ap^2^s','ap^2^d' };
% draw the results
for k =1:3
subplot(1,3,k),h1 =surf([rxx(:,:,k)]),hold on;h2 =surf([ryy(:,:,k)]);
    view( -37.5,5);set(gca,'FontSize',fontsize);colormap([1 1 1]);
    light('Position',[ -4 -4 1],'Style','Infinite');lightinggouraud;
    title(tname{ k });grid on;xlabel('r');ylabel('c');
    rxmin =min(min(rxx(:,:,k)));rymin =min(min(ryy(:,:,k)));
    text(1,1,rxmin -0.01,'r_x_x','FontSize',16);
    text(1,1,rymin +0.01,'r_x_a_x_a','FontSize',16);
end
hold off;
function [rxx,ryy] =getrxy(N,rs,C)
```

```
for k =1:3
    for i =1:N
        for j =1:N
            xn =rs(i* N:(i +1)* N -1,j* N:(j +1)* N -1);
            rxx(i,j,k) =max(max(xcorr2(xn,xn)));
            xnwo =rs((i -1)* N +1:(i +1)* N -1,(j -1)* N +1:(j +1)* N -1).* C(:,:,k);
            yn =[xnwo(N,N),xnwo(N,1:N -1) +xnwo(N,N +1:2* N -1);
                  xnwo(1:N -1,N) +xnwo(N +1:2* N -1,N),...
                  xnwo(1:N -1,1:N -1) +xnwo(N +1:2* N -1,N +1:2* N -1) +...
                  xnwo(1:N -1,N +1:2* N -1) +xnwo(N +1:2* N -1,1:N -1)];
                  ryy(i,j,k) =max(max(xcorr2(yn,yn)));
        end
    end
    mx =max(max(rxx(:,:,k)));
    rxx(:,:,k) =rxx(:,:,k)/mx;ryy(:,:,k) =ryy(:,:,k)/mx;
end
```

5.7 小结

本章首先引入二维正交变换基的概念，对常用的傅里叶、离散余弦和沃尔什变换的基本性质进行了总结并用 MATLAB 实现各种变换基图像；其次，构造了二维全相位信号处理的基本算法并实现原理框图，在对二维全相位信号处理流程分析的基础上给出优化算法；第三，分别基于 DFT、DCT 和 DWT 推导得到全相位变换公式以及求解全相位基和核的数学表达式，基于 MATLAB 又分别给出加窗情况下的特性对比效果三维图；最后，对二维全相位信号处理六个方面的特性进行了证明，这些性质对于 ap^2 的实际应用的挖掘将起到重要引导作用。

继续拓展到三维空间，全相位处理在时间维度上的表现有待挖掘。对于视频编码中的关键技术如参考帧形成、运动矢量估计以及时空内插滤波等方面，均是三维全相位变换应用点。

第 6 章 基于W-O的谱分析

对一个信号或系统进行分析有许多方法，大体上可以分成两类：一类是时域分析，另一类是频域分析，如图 6-1 所示。

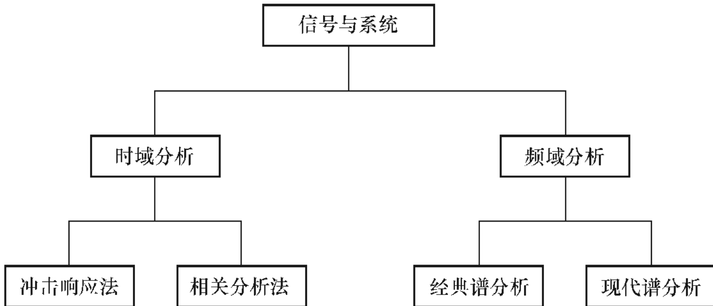


图 6-1　信号分析方法分类示意图

时域分析主要研究信号在时间域上的各种特性，如对确定性信号有冲击响应分析法，而对随机信号有相关分析法等。频域分析主要研究信号的各种频率特性。由于信号频率常常比原始信号的表达简单明了，如在解释随机信号的线性处理和滤波效应时，频谱表达容易得到简化，也可以清楚地看出信号被处理后特征的改变。因此谱分析是一种人们十分关注的重要方法。谱分析有很多类型，总的来说，可分为以傅里叶分析为理论基础的经典分析法和以非傅里叶分析为理论基础的现代分析法。

6.1　谱估计的一般理论

一般处理的信号可分为确定性信号和随机信号两类。确定性信号可用明确的数学关系来描述，通过变换方法所得到频域谱函数也是确定的。但对于随机信号，不同样本在同一时刻的值是不确定的。因此，即使对于某个样本记录可能存在着傅里叶变换，它还不能代表这个随机过程的样本总体的频谱。因此，只能对样本总体采用统计平均来描述。对于常见的各态历经的平稳随机信号，它的相关

函数是确定的，而相关函数与其频谱互为傅里叶变换对，这样就可以得到随机信号的功率谱密度。

6.1.1 经典功率谱估计

6.1.1.1 直接法

直接法又称周期图法，它是把随机信号 $x(n)$ 的 N 点观察数据 $x_N(n)$ 视为能量有限信号，直接取 $x_N(n)$ 的傅里叶变换得 $X_N(e^{j\omega})$，然后再取其幅值的平方并除以 N，作为对 $x(n)$ 真实功率谱 $S(e^{j\omega})$ 的估计。以 $\hat{S}_{PER}(e^{j\omega})$ 表示周期图法估计出的功率谱，则有

$$\hat{S}_{PER}(e^{j\omega})=\frac{1}{N}\left|X_N(e^{j\omega})\right|^2 \tag{6-1}$$

周期图这一概念是由 Schuster 于 1899 年首先提出的。对（6-1）式两边等间隔采样，得到其离散形式：

$$\hat{S}_{PER}(k)=\frac{1}{N}\left|X_N(k)\right|^2 \tag{6-2}$$

X_N（k）可以用 FFT 快速计算，所以 $\hat{S}_{PER}$（k）也可方便求出。

6.1.1.2 间接法

此方法的理论基础是维纳-辛钦定理。1958 年 Blackman 和 Tukey 给出了它的具体实现，即先由 $x_N(n)$ 估计出自相关函数 $\hat{R}(m)$，然后求 $\hat{R}$（m）的傅里叶变换，便得到 $x_N(n)$ 的功率谱，记为 $\hat{S}_{BT}(e^{j\omega})$，作为 $S(e^{j\omega})$ 的估计：

$$\hat{S}_{BT}(e^{j\omega}) = \sum_{m=-M}^{M}\hat{R}(m)e^{-jm\omega} \quad |M|\leqslant N-1 \tag{6-3}$$

因为由这种方法得到的功率谱是通过自相关函数间接得到的，所以称为间接法，又称 BT 法。它是在 FFT 问世之前常用的方法。

6.1.1.3 直接法的改进

直接法估计出的谱的方差特性不好，当数据长度 N 太大时，谱曲线起伏加剧，N 太小时，谱峰的分辨率又不好，因此需要加以改进。其中一种有效的方法是平均法，它的指导思想是把一长为 N 的数据 $x_N(n)$ 分成 L 段，分别求每一段的功率谱，然后予以平均，常用的方法有 Bartlett 法、Welch 法、Nuttall 法等。

1. Bartlett 法

将采样数据 $x_N(n)$ 分成 L 段，每段长度都是 M，第 i 段数据加矩形窗后为

$$x_N^i(n)=x_N[n+(i-1)M]d_1(n+(i-1)M) \quad 0\leqslant n\leqslant M-1,\ 1\leqslant i\leqslant L \tag{6-4}$$

式中，$d_1(n)$ 是长度为 M 的窗口。分别计算每一段的功率谱 $\hat{S}_{PER}^i(\omega)$：

$$\widehat{S}_{\mathrm{PER}}^{i}(\omega) = \frac{1}{M}\left|\sum_{n=0}^{M-1} x_{\mathrm{N}}^{i}(n)\mathrm{e}^{-\mathrm{j}n\omega}\right|^{2} \quad 1 \leqslant i \leqslant L \tag{6-5}$$

把 $\widehat{S}_{\mathrm{PER}}^{i}(\omega)$ 对应相加后再取平均即得到平均周期图 $\bar{S}_{\mathrm{PER}}(\omega)$ 为

$$\bar{S}_{\mathrm{PER}}(\omega) = \frac{1}{L}\sum_{i=1}^{L}\widehat{S}_{\mathrm{PER}}^{i}(\omega) = \frac{1}{ML}\sum_{i=1}^{L}\sum_{m=0}^{M-1}\left|x_{\mathrm{N}}^{i}(n)\mathrm{e}^{-\mathrm{j}m\omega}\right|^{2} \tag{6-6}$$

$\bar{S}_{\mathrm{PER}}(\omega)$ 的均值为

$$\begin{aligned} E\left\{\bar{S}_{\mathrm{PER}}(\omega)\right\} &= \frac{1}{L}\sum_{i=1}^{L}E\{\widehat{S}_{\mathrm{PER}}^{i}(\omega)\} = E\{\widehat{S}_{\mathrm{PER}}^{i}(\omega)\} \\ &= S(\omega) * \frac{1}{M}\left|D_1(\omega)\right|^2 = S(\omega) * W_1(\omega) \end{aligned} \tag{6-7}$$

式中，$D_1(\omega)$ 是矩形窗的频谱，$W_1(\omega)$ 是由矩形窗做相关所得到的长为 $2M-1$ 的三角窗频谱。显然，不取平均的周期图 $\widehat{S}_{\mathrm{PER}}(\omega)$ 和取平均后的 $\bar{S}_{\mathrm{PER}}(\omega)$ 都是有偏估计，且当 $N\to\infty$ 时，两者是渐近无偏的。

2. Welch 法

Welch 法是 Barlett 法的改进。第一个改进是对 $x_{\mathrm{N}}(n)$ 进行分段时，可允许每一段的数据有部分的交叠。例如，每一段数据重合一半时的段数为

$$L = \frac{N - M/2}{M/2} \tag{6-8}$$

式中，M 仍然是每段的长度，如图 6-2 所示。

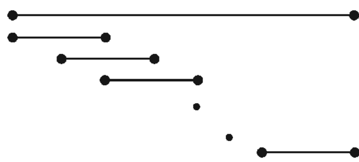


图 6-2　Welch 法的分段

第二个改进是每段的数据窗口可以不是矩形窗，例如使用汉宁窗或汉明窗，这样可以改善由于矩形窗旁瓣较大所产生的谱失真。这样得到的谱也是渐近无偏的，且由于允许重叠，使估计的方差得到很大的改善。

3. Nuttall 法

由于 Welch 法允许分段时重叠，这样就增加了段数 L，当然也就增加了做 FFT 的次数。如果用的数据窗是非矩形窗，这又大大增加了做乘法的次数。因此，Welch 法的计算量比较大。Nuttall 等人提出了一种五步结合算法，具体步骤如下：

步骤 1 和 2：与 Bartlett 法相同，即对 $x_{\mathrm{N}}(n)$ 自然分段（加矩形窗），且不交叠，得到平均后的功率谱 $\bar{S}_{\mathrm{PER}}(\omega)$。

步骤 3：由 $\bar{S}_{\mathrm{PER}}(\omega)$ 作反变换，得到该平均功率谱对应的自相关函数，记为

$\bar{R}(m)$，其最大宽度是 $2M-1$，$M=N/L$。

步骤 4：同间接法，对 $\bar{R}(m)$ 加延迟窗 $v_1(m)$，$v_1(m)$ 的最大单边宽度为 M，这样得到 $\bar{R}_{M_1}(m)$。

步骤 5：由 $\bar{R}_{M_1}(m)$ 作正变换，得到对 $x(n)$ 功率谱的估计，记为 $\bar{S}_{PBT}(\omega)$。

显然，这种方法是把直接法和间接法结合起来，同时把平滑和平均结合了起来，它既保持了平滑和平均减少方差的优点，而且计算量也较小。

上面 3 种改进方法可归纳为图 6-3。

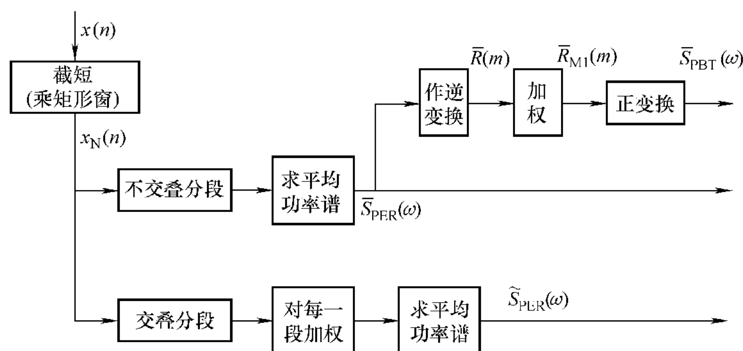


图 6-3　3 种改进方法的框图

古典谱估计，不论是间接法还是直接法，都可用 FFT 快速实现，且物理概念明确，因而仍是目前较常用的谱估计方法。但它有一些严重的缺点：

1）谱的分辨率低，它正比于 $2\pi/N$，N 是所使用的数据长度；

2）由于不可避免的窗函数影响，使得真正谱 $S(\omega)$ 在窗口主瓣内的功率向旁瓣部分“泄漏”，降低了分辨率。较大的旁瓣有可能掩盖 $S(\omega)$ 中较弱的成分，或是产生假的峰值。当分析的数据较短时，这些影响更为突出。

3）方差性不好，不是 $S(\omega)$ 的一致估计且 N 增大时谱曲线起伏加剧。

周期图的平滑和平均是和窗函数的使用紧紧相关联的。平滑和平均主要是用来改善周期图的方差性能，但往往又减小了分辨率和增大了方差。没有一个窗函数能使估计的谱在方差、偏差和分辨率各个方面都得到改善。因此，使用窗函数只是改进估计质量的一个技巧问题而不是根本的解决办法。

实现上述直接法、间接法和三种改进的直接法谱分析的 MATLAB 代码如下：

```
function spTA(N)

clc,clf;
```

```
if nargin = =0
  fs =24e3;f1 =2.1e3;f2 =8.3e3;
  t =0:1/fs:1;len_t =length(t);
  sigin =cos(2* pi* f1* t) +cos(2* pi* f2* t) +randn(size(t));
  N =2^nextpow2(length(sigin));M =1000;L =fix(len_t/M);
end
res1 =zjSP(sigin,fs,N);res2 =jjSP(sigin,fs,N);
res3 =btSP(sigin,fs,M);res4 =welchSP(sigin,fs,M);
res5 =nutalSP(sigin,fs,M);res ={res1,res2,res3,res4,res5};
tnm ={'direct','indirect','Bartlett','welch','nutal'};
nn =length(res);
for i =1:nn
   subplot(3,2,i),set(gca,'FontSize',16);
   resi =res{i};N =length(resi);
   h =plot(fs/N* [1:N/2]/1e3,10* log10(resi(1:N/2)));
   set(gca,'XLim',[0 10]);
   set(h,'color','k','LineWidth',2);grid on;
   xlabel('freq(kHz)');ylabel('psd(W/Hz)');title(tnm{i});
end
function res1 =zjSP(sigin,fs,N)
len_t =length(sigin);sigw =fft(sigin,N);
res1 =abs(sigw).^2/fs/len_t;

function res2 =jjSP(sigin,fs,N)
len_t =length(sigin);
Rxx =xcorr(sigin);srxx =[Rxx(len_t:end),fliplr(Rxx(1:len_t -1))];
res2 =fft(srxx,N)/fs/len_t;

function res3 =btSP(sigin,fs,M)
L =fix(length(sigin)/M);
for i =1:L
   sinput =sigin((i -1)* M +1:i* M);
   res(i,:) =zjSP(sinput,fs,M);
end
res3 =mean(res);
function res4 =welchSP(sigin,fs,M)
p =0.5;L =fix((length(sigin) -M* p)/(M -M* p));
ww =boxcar(M)';
for i =1:L
   sinput =sigin((i -1)* (M -M* p) +1:(i -1)* (M -M* p) +M);
   sinput =sinput.* ww;
```

```
    res(i,:)=zjSP(sinput,fs,M);
end
res4=mean(res);
function res5=nutalSP(sigin,fs,M)
m=2*M-1;
res=btSP(sigin,fs,m);
rx=ifft(res,m);
rxx=[rx((m+3)/2:end),rx(1:(m+1)/2)];
rxx=rxx.*boxcar(m)';
RXX=[rxx((m+3)/2:end),rxx(1:(m+3)/2)];
res5=fft(RXX,M)/fs/M;
```

运行结果如图 6-4 所示。

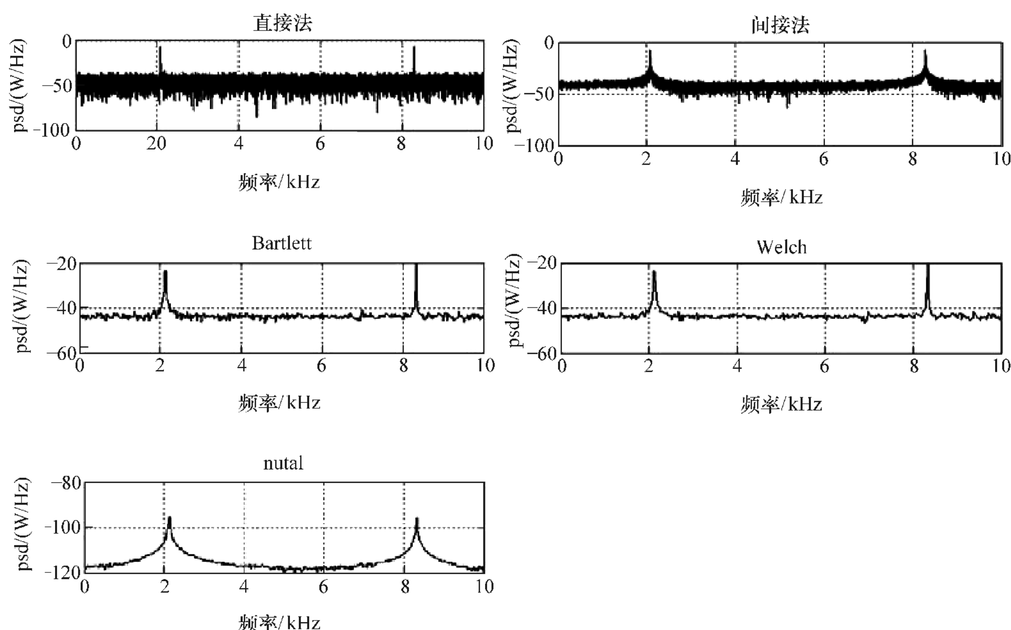


图 6-4 传统谱分析结果

图 6-4 中的 Welch 方法采用矩形窗加权并按照 50% 重叠，Nuttal 也采用矩形窗加权。观察图 6-4，采用 Nuttall 方法得到的谱图具有最优的平滑性但旁瓣泄漏也最大；直接法和间接法由于在整个信号区间进行分析，因此旁瓣影响最小，但噪声谱干扰也比较大，因此其方差最大。在频率分辨率方面，不论直接法和间接法还是分段法都使用相同的频率取样率 f_s 和数据长度（$N=LM$），所以各种方法具有相同的频率分辨率，但实际上考虑到分段时 L 的取整，Bartlett、Welch 和 Nuttall 三种方法的频率分辨率大于直接和间接法。

6.1.2　现代功率谱估计

经典谱分析是以对信号的傅里叶分析为理论基础的，而现代谱分析是建立在参数模型基础上的，它的主要特点是：①研究有限长度的信号或序列；②用统计的方法研究带有随机性信号的频谱特性；③与快速算法和 VLSI 技术密切结合，使新的理论和方法迅速在实际中得到应用。现代功率谱估计又称现代谱估计，但其强调这样一个事实：由于使用有限长度的信号而无法得到信号真实谱，所以各种新方法所得到的谱只能是真实谱的一种估计。

6.1.2.1　参数模型法

现代谱分析方法力图克服经典法的缺点，利用一些有关信号本身的知识，对被窗函数所截取的有限信号以外进行某种预测或外推，从而提高谱估计的质量。其中突出的优点就是提高了谱的分辨率和谱的真实程度。

模型法是现代谱估计的主要方法，模型法的共同之处是把一个具有许多变量的复杂过程化简为一个只用少量阐述就可以表示的简单过程。参数模型可分为两大类，一类是有理参数模型，这类模型可以用有理系统函数来表示，所求出的功率谱是连续的，也叫有理谱。它包括自回归（AR）模型（全极点模型）、滑动平均（MA）模型（全零点模型）和自回归滑动平均（ARMA）模型（极-零点模型）。另一类是指数模型，它假定信号模型为一些指数信号的线性组合，可以是衰减的指数信号、非衰减的指数信号或正弦信号的线性组合。用参数模型法进行谱估计首先要选择一种适合被测信号的参数模型，其次用适当的方法估计出模型的参数，最后利用所得到的模型参数计算出信号的谱。因此能否正确选择信号的模型，确定模型的阶数，以及估计模型的参数是决定参数模型法谱估计质量的关键。

实际中的许多随机离散时间信号 $x(n)$，都可以表示为有理参数模型。该模型的输入如果是一个方差为 σ^2 的白噪声序列 $u(n)$，输出为 $x(n)$，它们之间的关系可以用时域差分方程表示，即

$$x(n) = -\sum_{k=1}^{p} a_k x(n-k) + \sum_{i=0}^{q} b_i u(n-i) \tag{6-9}$$

或有理系统函数表示，即

$$H(z) = B(z)/A(z) \tag{6-10}$$

式中

$$B(z) = 1 + \sum_{i=1}^{q} b_i z^{-i} \quad A(z) = 1 + \sum_{k=1}^{p} a_k z^{-k} \tag{6-11}$$

该模型称为 ARMA 模型，此模型的功率谱为

$$P_{\mathrm{ARMA}}(f)=\frac{\sigma^2\ |B(\mathrm{e}^{\mathrm{j}2\pi f})|^2}{|A(\mathrm{e}^{\mathrm{j}2\pi f})|^2} \tag{6-12}$$

如果系数 $b_i=0$，$i=1$，2，…，q，则称这种模型为 AR 模型。如果系数 $a_k=0$，$k=1$，2，…，p，则称这种模型为 MA 模型。

有理系数模型中最常用的是 AR 模型。这种模型比较简单，且它是全极点模型，因此能较好地描述信号谱中的谱峰，表现出良好的频率分辨能力。AR 模型中包括许多不同的方法，它们的区别在于估计 AR 模型的参数所用的方法不同。利用线性预测方法，如果令前向线性预测误差为最小，可得到一组标准方程（Normal Equations），写成矩阵形式为

$$\begin{bmatrix}\varphi(0,0) & \varphi(1,0) & \cdots & \varphi(p,0)\\ \varphi(0,1) & \varphi(1,1) & \cdots & \varphi(p,1)\\ \vdots & \vdots & \ddots & \vdots\\ \varphi(0,p) & \varphi(1,p) & \cdots & \varphi(p,p)\end{bmatrix}\begin{bmatrix}1\\ a_1\\ \vdots\\ a_p\end{bmatrix}=\begin{bmatrix}\sigma^2\\ 0\\ \vdots\\ 0\end{bmatrix} \tag{6-13}$$

或表示成

$$\boldsymbol{\varphi}\cdot\boldsymbol{A}=\boldsymbol{\gamma} \tag{6-14}$$

若令

$$\varphi(i,j)=\frac{1}{N-p}\sum_{n=0}^{N-p-1}x^*(n-i)x(n-j) \tag{6-15}$$
$$i=0,1,\cdots,p;\ j=0,1,\cdots,p$$

则 $\varphi(i,j)=R(i-j)$，即为自相关函数。这时的式（6-13）称为尤拉-沃克方程，利用 Levinson 快速递推算法可以求出 AR 参数。如果令

$$\varphi(i,j)=\frac{1}{N-p}\sum_{n=p}^{N-1}x^*(n-i)x(n-j) \tag{6-16}$$
$$i=0,1,\cdots,p;\ j=0,1,\cdots,p$$

此时矩阵 $\boldsymbol{\varphi}$ 为协方差矩阵。可用侨莱斯基（Cholskey）分解法求解而得到 AR 参数。修正协方差可令前预测误差和后向预测误差的平均值为最小得到

$$\varphi(i,j)=\left[\sum_{n=p}^{N-1}x^*(n-i)x(n-j)+\sum_{n=0}^{N-1-p}x^*(n+i)x(n+j)\right]/[2(N-p)] \tag{6-17}$$

代入矩阵式（6-13）即可求解 AR 参数。这种方法用于估计含有多个十分靠近的正弦信号的频率时优于 AR 方法。

同古典谱估计方法相比，由于 AR 模型是一个有理分式，因而估计出的谱比古典谱估计法估计出的谱平滑。古典谱估计法中的谱分辨率正比于 $2\pi/N$，N 是数据长度。AR 模型谱的分辨率不受此限制，而主要取决于其极点接近单位圆的程度，每一个靠近单位圆的极点都将产生一个谱峰。在实际运用时，AR 模型在谱估计上存在一些缺点。一个明显的缺点是谱分辨率和 SNR 有着密切的关系。

SNR 低时，分辨率就会下降；其二，谱估计的质量受阶次 p 的影响，p 选得过低，谱太平滑，反映不出极点，p 选的过大，可以产生虚假的峰值；其三，如果 $x(n)$ 是含有噪声的正弦信号，峰值的位置易受 $x(n)$ 初相位的影响，且有的算法中出现“谱线分裂”现象，即在本应有的一个谱线的位置附近分裂成两个谱线。通过算法的改进和一些其他措施，可以较好地克服这些缺点。

在 MATLAB 中对尤拉-沃克的求解有四种方法，即自相关法、协方差法、修正协方差法和 Burg 法，对应的函数分别为 pyulear、pcov、pmcov 和 pburg，包含四个参数即随机信号序列 $\boldsymbol{x}$、AR 模型阶数 p、FFT 长度 $nfft$ 和取样率 fs，函数返回信号 $\boldsymbol{x}$ 的 AR 功率谱估计 P_{xx} 及谱范围 $\boldsymbol{\omega}$。假设平稳随机信号 $x(n)$ 构成如下：

$$x(n) = \sin(200n\pi) + 4\sin(220n\pi) + u(n) \tag{6-18}$$

实现上述 AR 模型现代谱估计的 MATLAB 代码如下：

```
functionspMD(N)

clc;clear;close all;
if nargin = =0
    N =1;
end
fs =1000;N =1;p =50;nfft =512;t =0:1/fs:N;
wn =randn(1,N* fs +1);
s1 =sqrt(20)* sin(2* pi* 100* t);
s2 =sqrt(2000)* sin(2* pi* 110* t);
x =s1 +s2 +wn;
x1 =xcorr(x,'biased');
funstr ={'pyulear(x1,p,nfft,fs)','pcov(x,p,nfft,fs)',...
    'pmcov(x,p,nfft,fs)','pburg(x,p,nfft,fs)'};
tname ={'xôïà¹ø·¨','Ð－·½²Î·¨','ÐЂPõýÐ－·½²Î·¨','Burg ·¨'};
for i =1:4
    [Pxx,f] =eval(funstr{i});
    subplot(2,2,i),set(gca,'FontSize',16);
    plot(f,10* log10(Pxx),'k','LineWidth',2);
    grid on;xlabel(' f(Hz)');ylabel('Pxx(ÍB/Hz)');title(tname{i});
end
```

运行结果如图 6-5 所示。

从图 6-5 可以看出，采用参数建模的谱估计方法得到的功率谱曲线平滑（方差小），分辨率高，可以明显地观察到两个谱峰。改变模型阶数 p，当降低至 20 时谱线平滑性增加但波峰分界消失，当增加至 100 时谱线振荡增加且出现波峰，以相关法为例的 MATLAB 运行结果如图 6-6 所示。

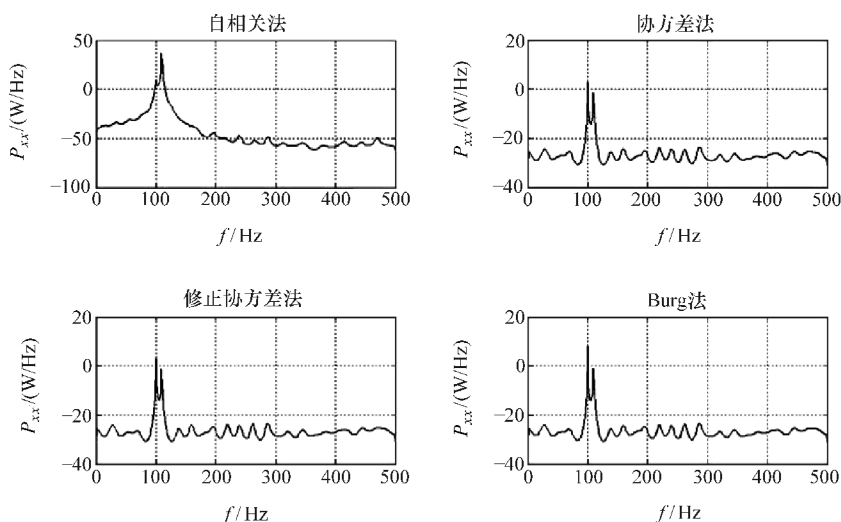


图 6-5　AR 模型现代谱分析结果

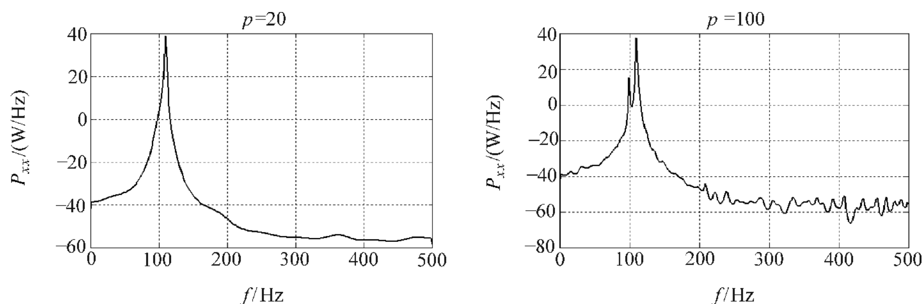


图 6-6　AR 模型阶数改变时谱分析结果

当阶数降低时，极点减少，因此曲线的平滑性更好；当阶数增大时，波峰增多，曲线的波动更多。

6.1.2.2　谱估计存在的问题

现代谱分析方法得到信号的频谱分辨率高，但对于其中大多数方法所表现出的统计性能的原因，很难从数学上给出解析式，因而也难以解释清楚。这不仅是由于方法本身的复杂性，而且因为对于有限数据的信号还没有成熟的统计分析的理论，因此对一些现代谱分析中容易混淆的问题应加以注意和区分。

频率分辨率的提高是现代谱分析方法的一个突出优点。如何衡量一种谱分析方法频率分辨能力的高低，通常是用实验的方法考察对两个振幅相等而频率十分靠近的正弦信号的分辨能力，用眼睛观察频谱曲线是否有两个对应的可分辨的尖峰。这种方式在谱分辨方法给定的情况下具有一定的参考价值但并不充分。

检测信号的能力另一个需要注意的问题，它是指从被噪声淹没的信号中检测出有用信号的过程。但选择什么样的谱估计方法取决于谱估计方法的应用场合，一种好的谱估计算法并不一定是好的信号检测算法。

6.2　全相位谱估计

6.2.1　基于 W-O 算法的谱估计

用 W-O 方法分析信号频谱是对 W-O 处理后的信号做傅里叶变换，这相当于信号首先经过一个线性系统 W-O，如图 6-7 所示。

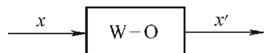


图 6-7　谱分析中的 W-O 模型

N 阶全相滤波器的输入矢量 $\boldsymbol{x}$ 是 $2N-1$ 阶的，图 6-7 中的输出等于 $\boldsymbol{x}'$ 与延迟窗 $\boldsymbol{C}$ 相乘后并重叠，即 $\boldsymbol{x}'$ 形式如下：

$$[c_0x_0+c_Nx_{-N}\quad c_1x_1+c_{N-1}x_{-N+1}\quad \cdots\quad c_{N-2}x_{N-2}+c_2x_{-2}\quad c_{N-1}x_{N-1}+c_1x_{-1}]^{\mathrm{T}} \tag{6-19}$$

设 $\boldsymbol{x}'=\boldsymbol{x}_0{}'+\boldsymbol{x}_1{}'$，其中

$$\begin{aligned}\boldsymbol{x}'_0&=[c_0x_0\quad c_1x_1\quad \cdots\quad c_{N-2}x_{N-2}\quad c_{N-1}x_{N-1}]^{\mathrm{T}}\\ \boldsymbol{x}'_1&=[c_Nx_{-N}\quad c_{N-1}x_{-N+1}\quad \cdots\quad c_2x_{-2}\quad c_1x_{-1}]^{\mathrm{T}}\end{aligned} \tag{6-20}$$

由式（6-20）得到 W-O 模型的输出 $\boldsymbol{x}'$ 频谱如下：

$$F_{x'}(\mathrm{j}\omega)=\frac{1}{2\pi}[\boldsymbol{C}_0(\mathrm{e}^{\mathrm{j}\omega})*\boldsymbol{X}_0(\mathrm{e}^{\mathrm{j}\omega})+\boldsymbol{C}_1(\mathrm{e}^{\mathrm{j}\omega})*\boldsymbol{X}_1(\mathrm{e}^{\mathrm{j}\omega})] \tag{6-21}$$

式中，$\boldsymbol{C}_0$ 和 $\boldsymbol{C}_1$ 分别定义如下：

$$\boldsymbol{C}_0(\mathrm{j}\omega)=\sum_{n=0}^{N-1}c_n\mathrm{e}^{-\mathrm{j}n\omega}\quad \boldsymbol{C}_1(\mathrm{j}\omega)=\sum_{n=0}^{N-1}c_{N-n}\mathrm{e}^{-\mathrm{j}n\omega} \tag{6-22}$$

分析具有单一频率 ω_0 的正弦信号的频谱，所以式（6-21）中两分量的频谱分别为

$$\begin{aligned}\boldsymbol{X}_0(\mathrm{j}\omega)&=\sum_{n=0}^{N-1}\boldsymbol{x}_n\mathrm{e}^{-\mathrm{j}n\omega}=\sum_{n=0}^{N-1}\mathrm{e}^{\mathrm{j}n(\omega_0-\omega)}=\frac{1-\mathrm{e}^{\mathrm{j}N(\omega_0-\omega)}}{1-\mathrm{e}^{\mathrm{j}(\omega_0-\omega)}}\\ \boldsymbol{X}_1(\mathrm{j}\omega)&=\sum_{n=0}^{N-1}\boldsymbol{x}_{-N+n}\mathrm{e}^{-\mathrm{j}n\omega}=\sum_{n=0}^{N-1}\mathrm{e}^{\mathrm{j}(-N+n)\omega_0}\mathrm{e}^{-\mathrm{j}n\omega}=\mathrm{e}^{-\mathrm{j}N\omega_0}X_0\end{aligned} \tag{6-23}$$

将式（6-21）代入式（6-23）得到：

$$F_{x'}(\mathrm{j}\omega)=\frac{1}{2\pi}[\boldsymbol{C}_0(\mathrm{j}\omega)+\boldsymbol{C}_1(\mathrm{j}\omega)\mathrm{e}^{-\mathrm{j}N\omega_0}]*\boldsymbol{X}_0 \tag{6-24}$$

由式（6-23）可以看出 $\boldsymbol{X}_0$ 就是经典频谱分析中信号加矩形窗时的频谱，因

此式（6-24）给出了经过 W-O 处理后信号 $\boldsymbol{x}'$ 频谱与原信号频谱 $\boldsymbol{x}$ 之间的关系。由数字信号处理基础理论可知，实际上在采用 DFT 对长度为 N 的截断信号进行谱计算时，$\boldsymbol{X}_0$ 只有在 $\omega=\omega_0+2k\pi/N$ 处的值正确，而采用 DFT 计算得到的各点频谱等于 $\boldsymbol{X}_0$ 在 $2n\pi/N(0\leqslant n\leqslant N-1)$ 处的抽样。即只有当 $\omega_0=2m\pi/N(m=n-k)$ 时运用 DFT 才得到正确的谱分析结果，频谱为在 m 处一个冲击（半个周期内）。由式（6-24）可得对应 $\boldsymbol{x}'$ 频谱如下：

$$F_{x'}(\mathrm{j}\omega)=\frac{1}{2\pi}[\boldsymbol{C}_0(\mathrm{j}\omega)+\boldsymbol{C}_1(\mathrm{j}\omega)]*\boldsymbol{X}_0 \tag{6-25}$$

对于单窗和无窗系统，由第 2 章中的内容可知其频偏向量 E_{N} 恒等于 **1**，所以结合式（6-22）和式（6-25）可得到 $\omega=\omega_0$ 时 $\boldsymbol{x}'$ 频谱为

$$\begin{aligned}F_{x'}(\mathrm{j}\omega)\big|_{\omega=\omega_0} &= \frac{1}{2\pi}\frac{1-\mathrm{e}^{-\mathrm{j}N\omega}}{1-\mathrm{e}^{\mathrm{j}\omega}}*\boldsymbol{X}_0=\left\{\frac{\mathrm{e}^{-\mathrm{j}\omega[(N-1)/2]}}{2\pi}\cdot\frac{\sin\frac{N\omega}{2}}{\sin\frac{\omega}{2}}\right\}*X_0(\mathrm{j}\omega)\\ &=\sum_{\omega=0}^{2\pi}\left\{\frac{\mathrm{e}^{-\mathrm{j}(\omega-\omega_0)[(N-1)/2]}}{2\pi}\cdot\frac{\sin\frac{N(\omega-\omega_0)}{2}}{\sin\frac{(\omega-\omega_0)}{2}}\right\}X_0(\mathrm{j}\omega_0)\\ &=\frac{N}{2\pi}X_0(\mathrm{j}\omega_0)\end{aligned} \tag{6-26}$$

即在各点处的频谱 $N/2\pi$ 倍于原信号频谱且未发生非线性转换，所以当 $N\geqslant8$ 时 $\boldsymbol{x}'$ 频谱将得到加强。对于双窗系统，E_{N} 不再恒等于 **1**，但具有“倒余弦”形状，这样 W-O 分析谱中在每点都产生旁瓣。由式（6-26）易知，这种“倒余弦”与 $\boldsymbol{X}_0$ 卷积的结果仍然保持了较大的主旁边能量比。

当 $\omega_0=2\pi(m+\theta)/N$（$0<\theta<1$）时，代入式（6-26）可得

$$F_{x'}(\mathrm{j}\omega_0)=\sum_{n=0}^{N-1}\left\{\frac{\mathrm{e}^{\mathrm{j}\pi(\theta-k)[(N-1)/N]}}{2\pi}\cdot\frac{\sin(\theta-k)\pi}{\sin\frac{(\theta-k)\pi}{N}}\right\}X_0(\mathrm{j}\omega)\quad k=m-n \tag{6-27}$$

由式（6-27）可以看出，由于非整数倍 θ 的存在，使 $\boldsymbol{x}'$ 在任意点的频谱都由原信号频谱全体通过加权得到，其权值函数 sin（·）/sin（·/N）的曲线具有突出主瓣且通过平均弱化旁瓣的特性。如 $N=32$，$\theta=0.3$，k 由 0 变化到 $N-1$ 时对应的曲线如图 6-8 所示。

由图 6-8 可以看出，当 $k=0$ 时即 $n=m$ 具有最大值，其余各点的权值有正有负而弱化了旁瓣的影响，使 $\omega=2\pi m/N$ 时谱线相对能量更大。由式（6-27）易知，$\boldsymbol{X}_0$ 在 $k=0$ 时的权值为 N，即阶数越高，W-O 谱线分辨力越强。如对频率 $f_0=12.3\mathrm{Hz}$ 的正弦信号以抽样频率 $f_{\mathrm{s}}=32\mathrm{Hz}$ 进行抽样并作 $N=32$ 的 FFT 谱分析，

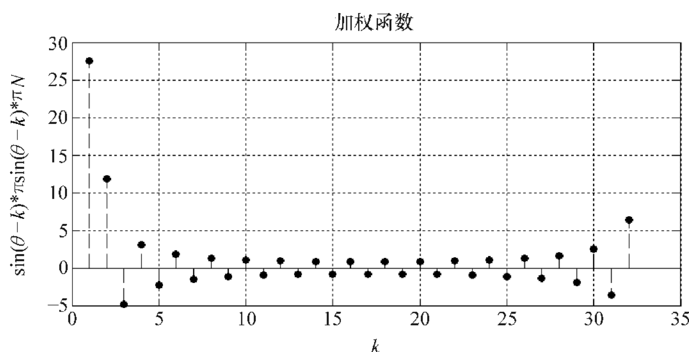


图 6-8　加权函数曲线

W-O 方法与传统加窗方法得到结果对比图如图 6-9 所示。

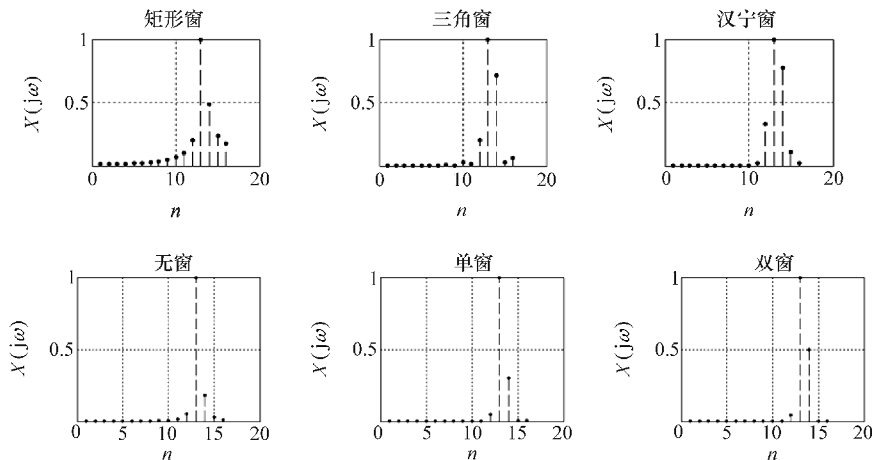


图 6-9　传统信号加窗与 W-O 谱分析

由图 6-9 可以看出，用 W-O 方法得到的信号频谱比用传统加窗方法具有更少、更低的旁瓣干扰。在无窗、单窗和双窗 W-O 分析中，双窗得到的频谱旁瓣最少，但第一旁瓣的干扰比无窗和单窗大而且对于整数倍单频的信号双窗 W-O 得到的频谱也有旁瓣。单窗比无窗 W-O 得到的信号频谱有较少的旁瓣但第一旁瓣干扰也较大。实现上面结果的 MATLAB 代码如下：

```
function spWO(N)


if nargin = =0
    f1 =12.3;fs =32;N =32;
end
clc,clf
```

```
t=0:1/fs:2;
sig=sin(2*pi*f1*t);

w(:,1)=boxcar(N);w(:,2)=triang(N);w(:,3)=hann(N);
wo(:,1)=conv(w(:,1),w(:,1));
wo(:,2)=conv(w(:,1),w(:,2));
wo(:,3)=conv(w(:,2),w(:,2));
wname={'矩形窗','三角窗','汉宁窗'};woname={'无窗','单窗','双窗'};

for k=1:3
    xx=sig(1:N)'.*w(:,k);
    X(:,k)=abs(fft(xx,N));X(:,k)=X(:,k)/max(X(:,k));
    subplot(2,3,k),set(gca,'FontSize',16);
    h=stem(X(1:N/2,k),'fill','k--');
    set(h,'MarkerSize',2);grid on;title(wname{k});
    xlabel('\itn');ylabel('\it\bfX\rm(j\it\omega)');

    yy=sig(1:2*N-1)'.*wo(:,k);
    xwo=[yy(N);yy(1:N-1)+yy(N+1:2*N-1)];

    Xwo(:,k)=abs(fft(xwo,N));Xwo(:,k)=Xwo(:,k)/max(Xwo(:,k));
    subplot(2,3,k+3),set(gca,'FontSize',16);
    h=stem(Xwo(1:N/2,k),'fill','k--');
    set(h,'MarkerSize',2);grid on;title(woname{k});
    xlabel('\itn');ylabel('\it\bfX’’\rm(j\it\omega)');
end
```

二维信号的 W-O 处理如图 5-26 所示，如果选择的卷积窗满足完全对称（即水平方向、垂直方向和中心都对称），则二维全相位 W-O 处理输出等于输入与 $\boldsymbol{C}$ 的卷积，从而有如下关系：

$$x'=x*\boldsymbol{C}\Rightarrow F_{x'}(\mathrm{j}\omega_r,\ \mathrm{j}\omega_c)=F_C(\mathrm{j}\omega_r,\ \mathrm{j}\omega_c)\cdot\boldsymbol{X}_0(\mathrm{j}\omega_r,\ \mathrm{j}\omega_c)\tag{6-28}$$

对于 ap^{2n}，卷积窗 $\boldsymbol{C}$ 的谱等于前后矩形基窗的乘积，容易推导得到：

$$F_C(\mathrm{j}\omega_r,\mathrm{j}\omega_c)=\left[\frac{\sin(N\omega_r/2)}{\sin(\omega_r/2)}\right]^2\left[\frac{\sin(N\omega_c/2)}{\sin(\omega_c/2)}\right]^2\mathrm{e}^{-\mathrm{j}(N-1)(\omega_r+\omega_c)}\tag{6-29}$$

结合式（6-28）和式（6-29），得到 ap^{2n}的 W-O 处理输出谱为

$$|F_{x'}(\mathrm{j}\omega_r,\mathrm{j}\omega_c)|=|\boldsymbol{X}_0(\mathrm{j}\omega_r,\mathrm{j}\omega_c)|\cdot\left[\frac{\sin(N\omega_r/2)}{\sin(\omega_r/2)}\right]^2\left[\frac{\sin(N\omega_c/2)}{\sin(\omega_c/2)}\right]^2\tag{6-30}$$

同一维结论相似，二维权值函数的曲线在行列方向上均具有突出主瓣且通过平均弱化旁瓣的特性。如 $N=32$ 的均匀取样得到的加权函数曲面如图 6-10 所示。

由曲面形状及式（6-30）可知，$\boldsymbol{x}'$ 谱线幅值是 $\boldsymbol{x}$ 在行列方向上的权值函数平方。主瓣的权值最大而旁瓣的权值按照平方关系快速衰减，这使 $\boldsymbol{x}'$ 谱线主瓣更加突出，所以 W-O 处理具有很好的抑制谱泄露特性。抑制效果与卷积窗形状密切相关，一种直观的改善方法是提高 sin（N·）/sin（·）的幂次。可通过两种途径来实现，其一是卷积窗形状，采用可分离基窗直接相乘得到二维基窗是常用的简易方法，生成的二维窗性质与一维基窗相同，但这种方法没有把行列方向上的互相关性考虑进去，因此，同时以行、列及 ±45°等方向上按照某种准则设计二维窗是值得探究的内容。其二是采用 $\text{ap}^{2\text{sd}}$ 方式进行 W-O 处理，在阶数相同条件下，幂次将分别提高 1 和 2。

图 6-10　二维 W-O 权值函数曲面

设二维信号由频率 $f_\mathrm{r}=12.3\text{Hz}$ 和 $f_\mathrm{c}=21.6\text{Hz}$ 的正弦信号组成且初相为零，即

$$s=\sin(2\pi f_\mathrm{r}t_\mathrm{r}+2\pi f_\mathrm{c}t_\mathrm{c}) \tag{6-31}$$

以抽样频率 $f_\mathrm{s}=64\text{Hz}$ 进行抽样并作 $N=32$ 的 FFT 谱分析，W-O 方法与传统加窗方法得到结果对比图如图 6-11 所示。

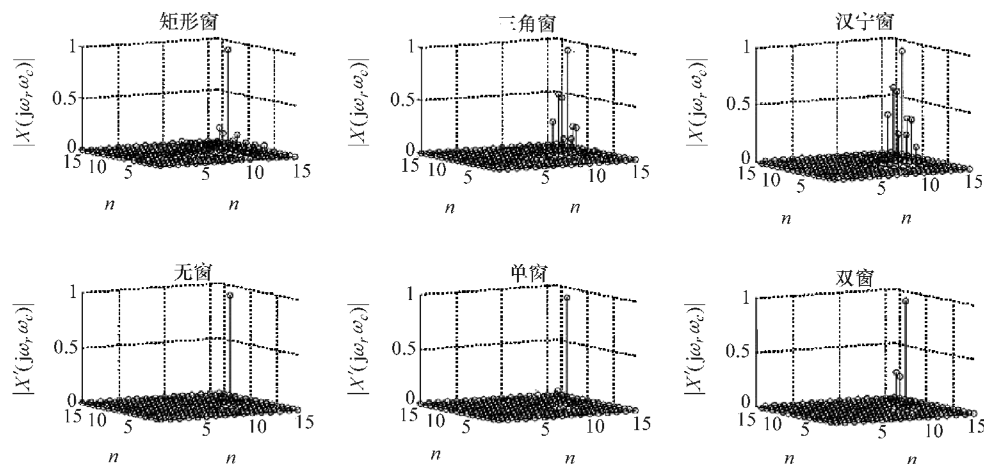


图 6-11　二维信号 W-O 谱分析结果对比

实现上述结果的 MATLAB 代码如下：

```
function spWO2(N)
if nargin==0
    fr=12.3;fc=21.6;fs=64;N=32;
    fname={'boxcar(N)','triang(N)','hann(N)'};
    index=['1','1','2','1','2','2'];
    wname={'矩形窗','三角窗','汉宁窗'};woname={'无窗','单窗','双窗'};
end
clc,clf
t=0:1/fs:2;n=length(t);
trs=repmat(t',1,n);tcs=repmat(t,n,1);

sig=sin(2*pi*fr*trs+2*pi*fc*tcs);

for k=1:3
    x=eval(fname{k});w2(:,:,k)=x*x';
    y1=w2(:,:,str2num(index(k)));y2=w2(:,:,str2num(index(k+3)));
    wo2(:,:,k)=conv2(y1,y2);
    wo2(:,:,k)=wo2(:,:,k)/max(max(wo2(:,:,k)));
end
for k=1:3
    xx=sig(1:N,1:N)'.*w2(:,:,k);
    X(:,:,k)=abs(fft2(xx,N,N));X(:,:,k)=X(:,:,k)/max(max(X(:,:,k)));
    subplot(2,3,k),set(gca,'FontSize',16);
    h=stem3(X(1:N/2,1:N/2,k)','k');view(-30,10);box off;
    set(gca,'XLim',[1,N/2],'YLim',[1,N/2]);grid on;
    title(wname{k});xlabel('\itn');ylabel('\itn');
    zlabel('|\it\bfX\rm(j\it\omega_r\rm,\it\omega_c\rm)|');

    yy=sig(1:2*N-1,1:2*N-1).*wo2(:,:,k);
    xwo=[yy(N,N),yy(N,1:N-1)+yy(N,N+1:2*N-1);...
        yy(1:N-1,N)+yy(N+1:2*N-1,N),...
        yy(1:N-1,1:N-1)+yy(N+1:2*N-1,N+1:2*N-1)+...
        yy(1:N-1,N+1:2*N-1)+yy(N+1:2*N-1,1:N-1)];
    Xwo(:,:,k)=abs(fft2(xwo,N,N));Xwo(:,:,k)=Xwo(:,:,k)/max(max(Xwo(:,:,
k)));
    subplot(2,3,k+3),set(gca,'FontSize',16);
    stem3(Xwo(1:N/2,1:N/2,k),'k');view(-30,10);box off;
    set(gca,'XLim',[1,N/2],'YLim',[1,N/2]);grid on;
    title(woname{k});xlabel('\itn');ylabel('\itn');
    zlabel('|\it\bfX''\rm(j\it\omega_r\rm,\it\omega_c\rm)|');
end
```

6.2.2　W-O 谱分析在信号检测中的应用

为了检测 W-O 算法对信号检测能力的提高，要对含有噪声的双频正弦波做谱分析。设信号由频率分别为 240Hz 和 248.8Hz，幅值为 0.3 的正弦波组成，并受均值为 0，方差为 1 的高斯白噪声 $n(t)$ 干扰，信噪比为 −10dB，信号被噪声淹没，即

$$s(t)=0.3\sin(2\pi\cdot 240t)+0.3\sin(2\pi\cdot 248.8t)+n(t) \tag{6-32}$$

采样频率 $f_s=2048\text{Hz}$，进行长度为 512 点的 FFT 仿真结果如图 6-12 所示（取半边谱图）。

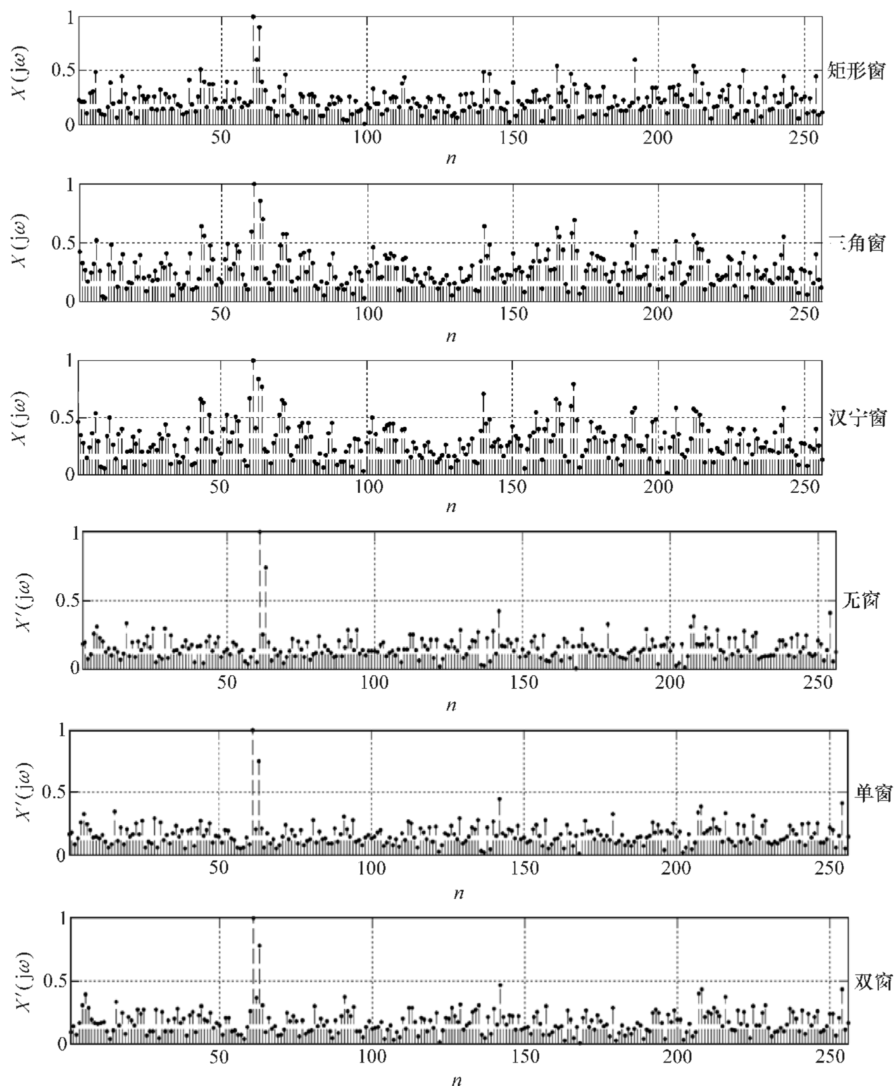


图 6-12　一维信号 W-O 谱检测对比图

传统加矩形窗、三角窗和汉宁窗与 ap^{nsd} W-O 谱检测结果中两个正弦相对幅值及噪声谱统计结果见表 6-1。

表 6-1 一维信号 W-O 谱检测对比统计结果

	240Hz 幅度	248.8Hz 幅度	干扰谱幅度	噪声谱均值	噪声谱方差
矩形窗	1.00	0.90	0.60	0.2256	0.1556
三角窗	1.00	0.85	0.60/0.70	0.2705	0.1620
汉宁窗	1.00	0.84	0.67/0.76	0.2923	0.1722
ap^{n} W-O	1.00	0.74	0.25	0.1302	0.0970
ap^{s} W-O	1.00	0.75	0.21	0.1503	0.0988
ap^{d} W-O	1.00	0.78	0.36	0.1609	0.1073

上面的实验结果验证了频率相近的两个信号通过 W-O 谱分析后分离度得到增加，便于提取。W-O 谱提升了信号能量，使邻近噪声谱得到很大程度的压抑。无论从谱均值还是方差来对比，W-O 效果均优于传统直接分析法。MATLAB 实现代码与上述 spWO 函数基本相同，不再列出。需要注意是，MATLAB 中噪声通过随机函数来生成，其中 rand 产生的噪声满足均匀分布，randn 则是产生满足均值为 0 和方差为 1 的正态分布即高斯白噪声。

二维信号 W-O 谱分析进行信号检测的过程与一维信号类似，假设二维信号 s 成分中信噪比为 -10dB，构成如下：

$$s=0.3\sin(480\pi t_r+497.6\pi t_c)+n(t_r,\ t_c) \tag{6-33}$$

式中，$n(t_r,\ t_c)$ 是均值为 0，方差为 1 的二维高斯白噪声。采样频率 $f_s=1024$Hz，进行长度为 64 点的二维 FFT 谱分析，W-O 方法与传统加窗方法得到结果对比图如图 6-13 所示。

统计结果见表 6-2。

表 6-2 二维信号 W-O 谱检测对比统计结果

	240Hz 幅度	248.8Hz 幅度	干扰谱幅度	噪声谱均值	噪声谱方差
矩形窗	1.00	0.94	0.43	0.1338	0.0790
三角窗	1.00	0.96	0.63/0.52	0.1358	0.0651
汉宁窗	1.00	0.96	0.72/0.63/0.56	0.1413	0.0750
ap^{2n} W-O	1.00	0.44	0.31	0.1050	0.0559
ap^{2s} W-O	1.00	0.53	0.28	0.1256	0.0596
ap^{2d} W-O	1.00	0.64	0.29	0.1180	0.0696

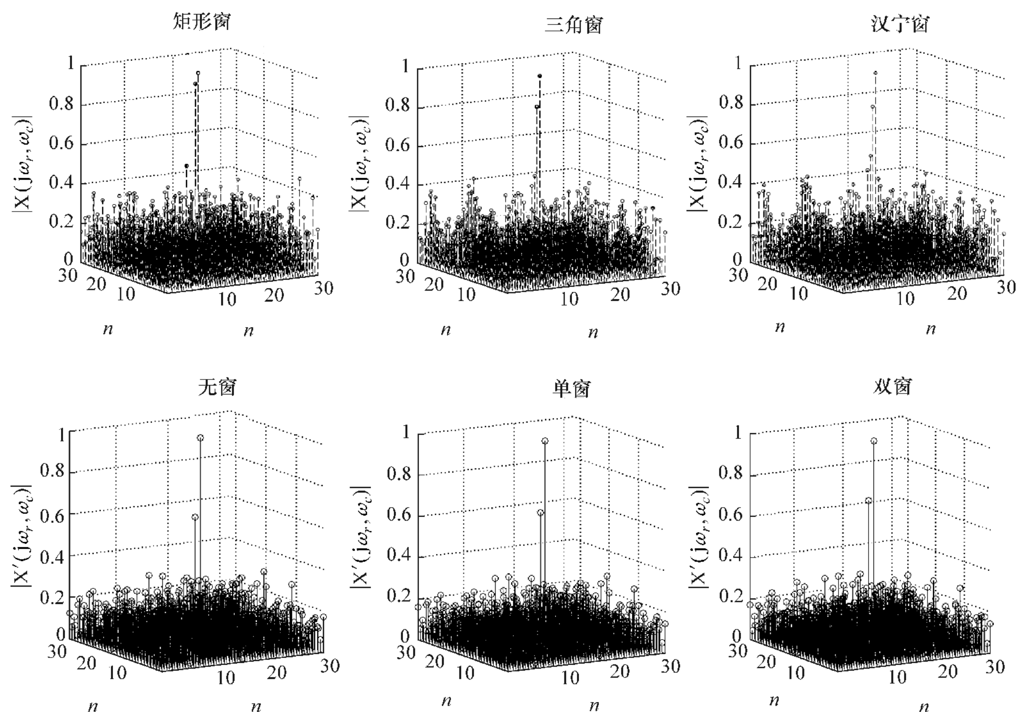


图 6-13　二维信号 W-O 谱检测对比图

6.3　小结

本章首先介绍了谱分析的两种常用方法：经典谱估计和现代谱估计。对经典法的直接法和间接法以及直接法的 3 种改进方案进行了简单的说明。此外，还介绍了现代谱估计中的 AR 模型，分别指出它们存在的优缺点。然后，给出基于 W-O 的谱估计思想，而且从理论上给予了较为详尽的说明。最后，通过对一维和二维信号谱检测实验，验证了 W-O 算法的可行性和优点，并列出全部 MATLAB 代码供读者深入研究参考。

第 7 章 设计滤波器组的全相位方法

20 世纪 70 年代初，滤波器组首先在语音编码和多路复用中得到了应用。其后，由于多速率信号处理理论的发展和子带编码技术的广泛应用，滤波器组理论在 30 多年的时间里得到了充分发展，许多新的理论和设计方法不断涌现。

所谓滤波器组就是将信号按不同的频带分解成一组信号，并将分解得到的子带信号重新合成原信号的处理系统。将一个信号分解成不同频带信号的过程被称为分析过程；反之，将不同频带的信号合成为一个信号的过程被称为综合过程。我们知道，滤波器组中的误差通常来自 4 个方面，即由于抽取和内插所产生的混叠和镜像是误差源之一，这种误差所产生的失真称为混叠失真；由滤波器幅频特性的波纹产生的误差所带来的失真称为幅度失真；由滤波器的相频特性的非线性而产生的误差所带来的失真叫相位失真；由编解码所产生的非线性失真称为子带量化误差，这是一种无法完全消除的误差。对如何消除这些失真人们已做了大量研究，其中完全消除混叠失真是个重要任务。最初有关滤波器组的研究多集中在正交镜像及半边带 FIR 滤波器组的设计和实现，采用窗函数或时域法来实现，但这些方法很难控制滤波器组的重构误差。由 T. P. Barnwell 等人提出的用 IIR 实现 QMFB（Qurature Multirate Filter Bank，多速率正交滤波器组）的方法也存在不同的相位和幅度失真。80 年代后期，M. J. Smith 等人提出了运用谱分解法设计无失真地重建原始信号的滤波器方法，但存在相位失真。

7.1 抽取和内插

7.1.1 一维信号分析

多率系统中最基本的运算就是抽取和内插，目的是为了满足各处需要不同的抽样率，以利于信号的简化运算处理、编码、传输和存储。使抽样率降低的运算称为抽取，实现此转换的处理单元称为抽取器；使抽样率升高的运算称为内插，对应的单元称为内插器。抽取因子为 M 的抽取器和内插因子为 L 的内插器如图 7-1所示。图 7-1 中，$x(n)$ 是输入信号，$y_D(n)$ 和 $y_E(n)$ 是抽取器和内插器的输出，M 和 L 是整数。抽取器的输入输出关系表达式如下：

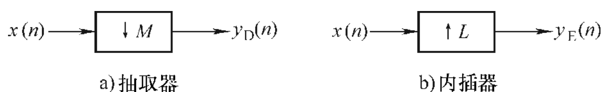


图 7-1　抽取器和内插器

$$y_D(n) = x(Mn) \tag{7-1}$$

即信号通过抽取器后，只有在 M 的整数倍时刻点被保留下来。例如 $M=2$ 时抽取后的 $x(n)$ 和 $y_D(n)$ 的关系如图 7-2 所示。

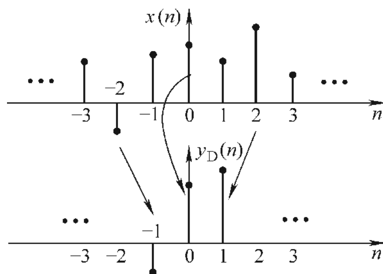


图 7-2　$M=2$ 时抽取后的 $x(n)$ 和 $y_D(n)$ 的映射关系

内插器的输入输出关系表达式如下：

$$y_E(n) = \begin{cases} x(n/L) & n/L \in \mathrm{Z} \\ 0 & 其他 \end{cases} \tag{7-2}$$

例如 $L=2$ 时内插后的 $x(n)$ 和 $y_E(n)$ 的关系如图 7-3 所示。

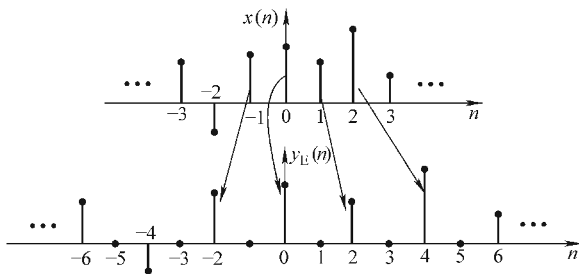


图 7-3　$L=2$ 时内插后的 $x(n)$ 和 $y_E(n)$ 的映射关系

内插器是通过 $x(n)$ 相邻点之间通过加 $L-1$ 个 0 的方式得到 $y_E(n)$ 即内插器没有损失任何信息，所以 $x(n)$ 可以利用 $y_E(n)$ 完全恢复。从频谱上来分析，内插器的输入和输出关系为

$$Y_E(z)=\sum_{n=-\infty}^{+\infty}y_E(n)z^{-n}=\sum_{k=-\infty}^{+\infty}y_E(kL)z^{-kL}=\sum_{k=-\infty}^{+\infty}x(k)z^{-kL}=X(z^L)$$
$$\Rightarrow Y_E(e^{j\omega})=X(e^{j\omega L}) \tag{7-3}$$

式（7-3）说明，内插后的信号带宽压缩为原来的 $1/L$，周期为 $2\pi/L$。所以，在 $[0,2\pi)$ 间除了在 0 点处的压缩谱外，还在 $(0,2\pi)$ 间产生 $L-1$ 个镜像谱。谱关系如图 7-4 所示。

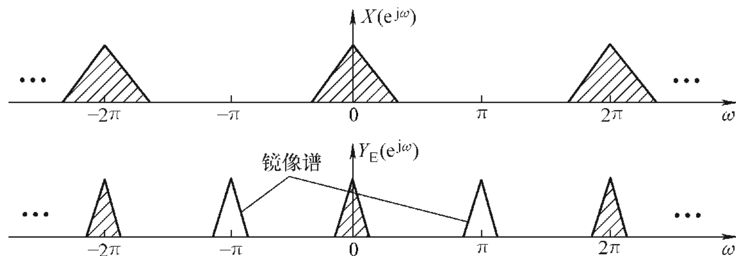


图 7-4　$L=2$ 时内插器的频谱特性

抽取器的输入和输出谱关系推导如下：

$$Y_D(z)=\sum_{n=-\infty}^{+\infty}y_D(n)z^{-n}=\sum_{n=-\infty}^{+\infty}x(nM)z^{-n}=\sum_{k=-\infty}^{+\infty}x(k)z^{-k/M}\quad k=nM \tag{7-4}$$

定义梳状序列 $C_M(n)$ 如下：

$$C_M(k)=\begin{cases}1 & k=nM,\ k\in Z\\ 0 & 其他\end{cases} \tag{7-5}$$

则有

$$x(k)=C_M(n)x(n)\quad k=nM,\ n\in(-\infty,+\infty) \tag{7-6}$$

梳状序列可以表示为

$$C_M(n)=\frac{1}{M}\sum_{k=0}^{M-1}W_M^{-kn}W_M=e^{-j2\pi/M} \tag{7-7}$$

将式（7-6）和式（7-7）代入到式（7-4）得到：

$$Y_D(z)=\frac{1}{M}\sum_{n=-\infty}^{+\infty}\sum_{k=0}^{M-1}x(n)\ (z^{1/M}W^k)^{-n}=\frac{1}{M}\sum_{k=0}^{M-1}X(z^{1/M}W^k)$$
$$\Rightarrow Y_D(e^{j\omega})=\frac{1}{M}\sum_{k=0}^{M-1}X(e^{j(\omega-2k\pi)/M}) \tag{7-8}$$

式（7-8）表明，抽取后的频谱等于原信号频谱膨胀 M 倍后再分别平移 $2k\pi$ 得到的 M 个谱之和，幅度则为原来的 $1/M$。$M=2$ 时抽取后谱关系如图 7-5 所示。可以看出，因为 $X(e^{j\omega})$ 的截止频率 $\omega_c>\pi/2$，所以 $Y_D(e^{j\omega})$ 发生了重叠。

为避免频谱重叠，要求 $x(n)$ 为低通带限信号且满足 $|\omega_c|\leqslant\pi/M$。在应用

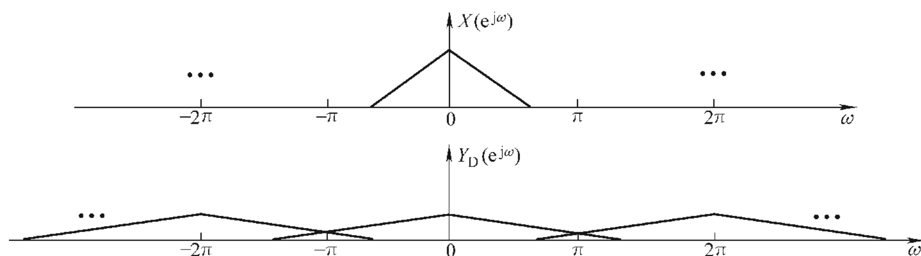


图 7-5　$M=2$ 抽取器频谱特性

中，抽取器前往往连接一个低通滤波器，称之为抽取滤波器，其频率特性如图 7-6b所示。完成的抽取过程如图 7-6a 所示。

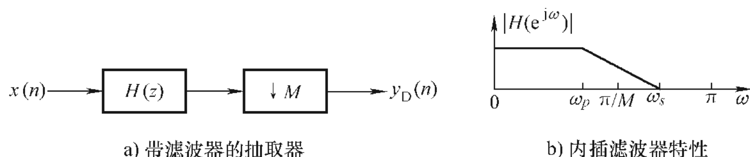


图 7-6　带内插滤波器的内插器及滤波器特性

滤波器的作用是消除混叠误差。同样，应用中的内插器后面常连接一个低通内插滤波器，其主要作用是消除频谱镜像，完整的内插电路和滤波器特性如图 7-7所示。

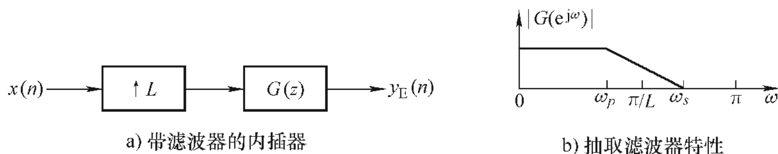


图 7-7　带抽取滤波器的抽取器及滤波器特性

利用 MATLAB 实现 $M=4$ 的抽取和 $L=2$ 的内插过程及谱分析代码如下：

```
   function specDI(N)
if nargin = =0
   N =128;M =4;L =2;
end
clc,clf
t = -1:2/N:1;t(t = =0) =eps;x =sin(10* t)./(t);x =x.* x;
dx =decS(x,M);getSP(x,dx,N,0);
ix =intS(dx,L);getSP(dx,ix,N,1);
```

```
function dx = decS(x,M)
n = length(x);N = fix(n/M);
dx = x(1:M:N* M);figure(1);
subplot(1,2,1),set(gca,'FontSize',16);
stem(x,'k--','fill','MarkerSize',2);grid on;
xlabel('\itn');ylabel('\itx\rm(\itn\rm)');title('original signal');
subplot(1,2,2),set(gca,'FontSize',16);
stem(dx,'k--','fill','MarkerSize',2);grid on;xlabel('\itn');
ylabel('\ity_D\rm(\itn\rm)');title('after decimation \itM\rm=4');
function ix = intS(x,L)
n = length(x);N = L* n;ix = zeros(1,N);
ix(1:L:N-L+1) = x;figure(3);
subplot(1,2,1),set(gca,'FontSize',16);
stem(x,'k--','fill','MarkerSize',2);grid on;xlabel('\itn');
ylabel('\ity_D\rm(\itn\rm)');title('original signal');
subplot(1,2,2),set(gca,'FontSize',16);
stem(ix,'k--','fill','MarkerSize',2);grid on;xlabel('\itn');
ylabel('\ity_E\rm(\itn\rm)');title('after interpolation \itL\rm=2');

function p = getSP(sig,dx,N,flag)
w = 0:2* pi/N:pi;nw1 = [0:length(sig)-1]'* w;
xs1 = repmat(sig',1,size(nw1,2));res1 = abs(sum(exp(-j* nw1).* xs1));

nw2 = [0:length(dx)-1]'* w;xs2 = repmat(dx',1,size(nw2,2));
res2 = abs(sum(exp(-j* nw2).* xs2));
if flag = =0
    figure(2);ystr1 = '\itX\rm(\ite^j^\omega\rm)';
    tstr1 = '\it\bfX spectrum';ystr2 = '\itY_D\rm(\ite^j^\omega\rm)';
    tstr2 = '\it spectrum after decimation';
else
    figure(4);ystr1 = '\itY_D\rm(\ite^j^\omega\rm)';
    tstr1 = '\it\bfY_D spectrum';ystr2 = '\itY_E\rm(\ite^j^\omega\rm)';
    tstr2 = '\it spectrum after interpolation';
end
subplot(1,2,1),set(gca,'FontSize',16);
stem(res1,'k--','fill','MarkerSize',2);grid on;
xlabel('\itn');ylabel(ystr1);title(tstr1);
subplot(1,2,2),set(gca,'FontSize',16);
stem(res2,'k--','fill','MarkerSize',2);grid on;
xlabel('\itn');ylabel(ystr2);title(tstr2);
```

实验结果如图 7-8 所示。

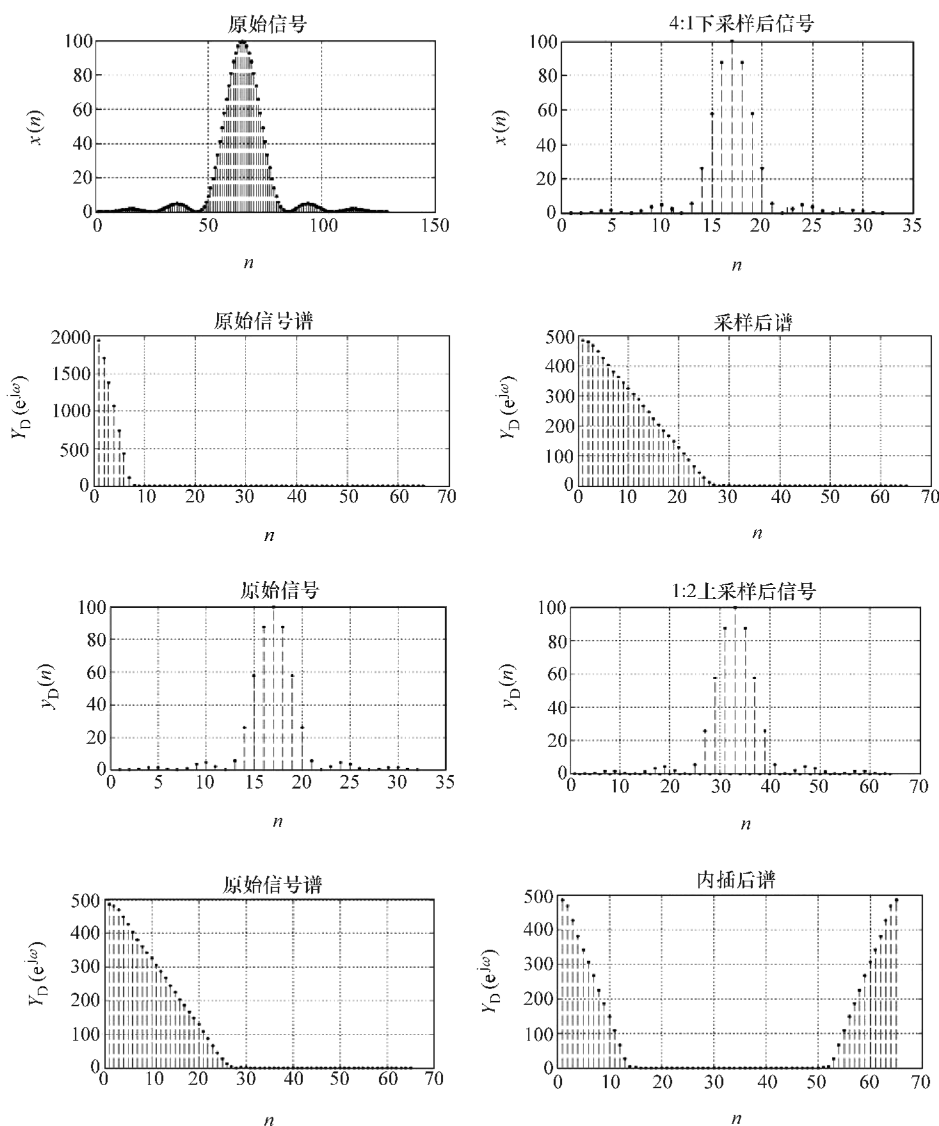


图 7-8　一维信号抽取和内插实验

7.1.2　二维信号分析

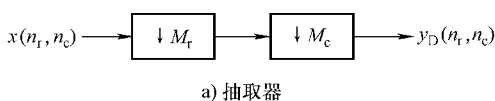
二维信号的抽取和内插分别在行和列方向上进行，以 M_r 和 M_c 为行列抽取因子的抽取器及以 L_r 和 L_c 为行列内插因子的内插器如图 7-9 所示。

图中，$x(n_r, n_c)$ 是输入信号，$y_D(n_r, n_c)$ 和 $y_E(n_r, n_c)$ 是抽取器和内插

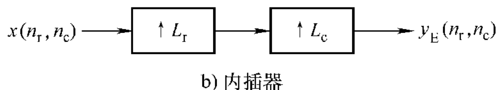
器的输出，M_r、M_c 和 L_r、L_c 均是整数。抽取器的输入输出关系表达式如下：

$$y_D(n_r, n_c) = x(M_r n_r, M_c n_c) \tag{7-9}$$

即信号通过抽取器后，行方向上只有在 M_r 的整数倍以及列方向上在 M_c 的整数倍时刻点被保留下来。例如$(M_r, M_c)=(3, 2)$时的抽取如图 7-10 所示。



a) 抽取器



b) 内插器

图 7-9　二维抽取器和内插器

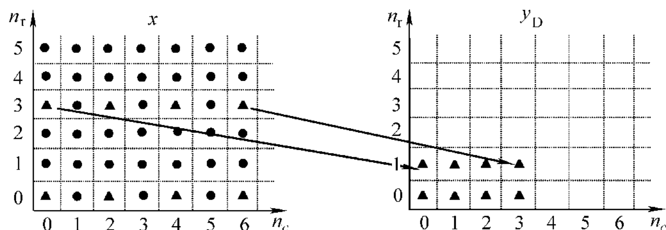


图 7-10　$(M_r, M_c)=(3, 2)$二维抽取器 $\boldsymbol{x}$ 和 $\boldsymbol{y}_D$ 的映射关系

二维内插器的输入输出关系表达式如下：

$$y_E(n_r, n_c) = \begin{cases} x(n_r/L_r, n_c/L_c) & n_r/L_r, n_c/L_c \in Z \\ 0 & 其他 \end{cases} \tag{7-10}$$

例如$(L_r, L_c)=(2, 3)$时的二维内插如图 7-11 所示。

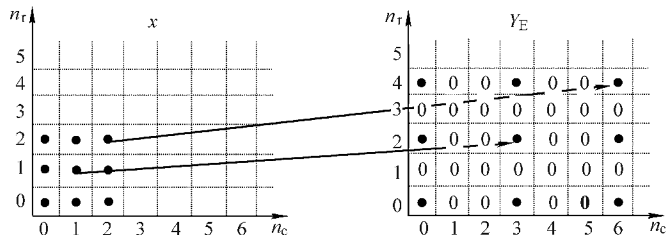


图 7-11　$(L_r, L_c)=(2, 3)$二维内插器 $\boldsymbol{x}$ 和 $\boldsymbol{y}_E$ 的映射关系

从频谱上来分析，二维内插器的输入和输出关系为

$$\begin{aligned} Y_E(z_r, z_c) &= \sum_{n_r=-\infty}^{+\infty}\sum_{n_c=-\infty}^{+\infty} y_E(n_r, n_c) z_r^{-n_r} z_c^{-n_c} = \sum_{k_r=-\infty}^{+\infty}\sum_{k_c=-\infty}^{+\infty} y_E(k_r L_r, k_c L_c) z_r^{-k_r L_r} z_c^{-k_c L_c} \\ &= \sum_{k_r=-\infty}^{+\infty}\sum_{k_c=-\infty}^{+\infty} x(k_r, k_c) z_r^{-k_r L_r} z_c^{-k_c L_c} = X(z^{L_r}, z^{L_c}) \\ &\Rightarrow Y_E(e^{j\omega_r}, e^{j\omega_c}) = X(e^{j\omega_r L_r}, e^{j\omega_{cr} L_c}) \end{aligned} \tag{7-11}$$

式（7-11）说明，内插后的频带在行方向压缩为原来的 $1/L_r$ 而列方向上压缩为 $1/L_c$，即总体带宽为原来的 $1/L_rL_c$。所以，在 $[0, 2\pi)\times[0, 2\pi)$ 二维空间除了在（0，0）点处的压缩谱外，还在 $(0, 2\pi)\times(0, 2\pi)$ 间产生 L_rL_c-1 个镜像谱。谱关系如图 7-12 所示。

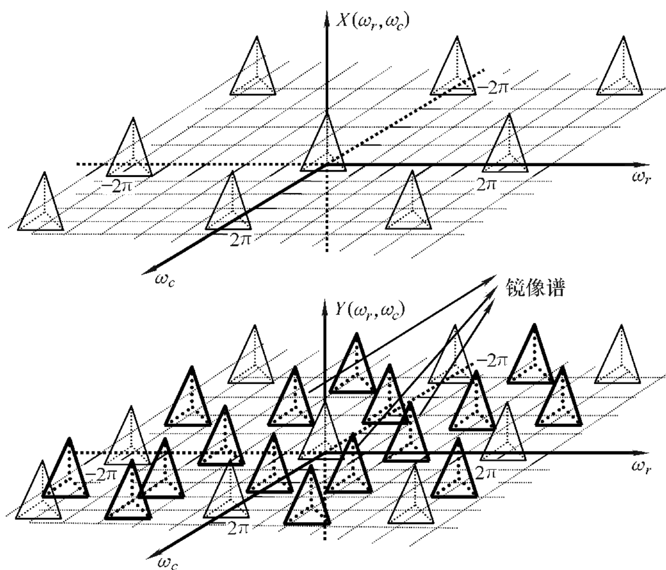


图 7-12　$(L_r,L_c)=(2,2)$ 二维内插器频谱特性

二维抽取器的输入和输出谱关系推导如下：

$$
\begin{aligned}
Y_D(z_r,z_c) &= \sum_{n_r=-\infty}^{+\infty}\sum_{n_c=-\infty}^{+\infty} y_D(n_r, n_c) z_r^{-n_r} z_c^{-n_c} = \sum_{n_r=-\infty}^{+\infty}\sum_{n_c=-\infty}^{+\infty} x(n_rM_r, n_cM_c) z_r^{-n_r} z_c^{-n_c} \\
&= \sum_{k_r=-\infty}^{+\infty}\sum_{k_c=-\infty}^{+\infty} x(k_r, k_c) z_r^{-k_r/M_r} z_c^{-k_c/M_c}, \quad k_r = n_rM_r,\ k_c = n_cM_c
\end{aligned} \tag{7-12}
$$

定义梳状序列 $C_{M_r,M_c}(n_r, n_c)$ 如下：

$$
C_{M_r,M_c}(k_r, k_c) = \begin{cases} 1 & k_r = n_rM_r,\ k_c = n_cM_c, \quad k_r, k_c \in Z \\ 0 & \text{其他} \end{cases} \tag{7-13}
$$

则有

$$
x(k_r, k_c) = C_{M_r,M_c}(n_r, n_c)x(n_r, n_c), \quad k = nM, \quad n_r, n_c \in (-\infty, +\infty) \tag{7-14}
$$

梳状序列可以表示为

$$
C_{M_r,M_c}(n_r, n_c) = \frac{1}{M_r}\sum_{k=0}^{M_r-1} W_{M_r}^{-kn_r} \cdot \frac{1}{M_c}\sum_{k=0}^{M_c-1} W_{M_c}^{-kn_c},\ W_{M_r} = e^{-j2\pi/M_r},\ W_{M_c} = e^{-j2\pi/M_c} \tag{7-15}
$$

将式（7-14）和式（7-15）代入到式（7-12）得到：

$$
\begin{aligned}
Y_{\mathrm{D}}(z_{\mathrm{r}}, z_{\mathrm{c}}) &= \frac{1}{M_{\mathrm{r}}M_{\mathrm{c}}}\sum_{k_{\mathrm{r}}=0}^{M_{\mathrm{r}}-1}\sum_{k_{\mathrm{c}}=0}^{M_{\mathrm{c}}-1}\left[\sum_{n_{\mathrm{r}}=-\infty}^{+\infty}\sum_{n_{\mathrm{c}}=-\infty}^{+\infty}x(n_{\mathrm{r}}, n_{\mathrm{c}})(z_{\mathrm{r}}W_{\mathrm{r}}^{k_{\mathrm{r}}})^{-n_{\mathrm{r}}}(z_{\mathrm{c}}W_{\mathrm{c}}^{k_{\mathrm{c}}})^{-n_{\mathrm{c}}}\right] \\
&= \frac{1}{M_{\mathrm{r}}M_{\mathrm{c}}}\sum_{k_{\mathrm{r}}=0}^{M_{\mathrm{r}}-1}\sum_{k_{\mathrm{c}}=0}^{M_{\mathrm{c}}-1}X(z_{\mathrm{r}}^{1/M_{\mathrm{r}}}W_{\mathrm{r}}^{k_{\mathrm{r}}}, z_{\mathrm{c}}^{1/M_{\mathrm{c}}}W_{\mathrm{c}}^{k_{\mathrm{c}}})
\end{aligned}
\tag{7-16}
$$

$$
\Rightarrow Y_{\mathrm{D}}(\mathrm{e}^{\mathrm{j}\omega_r}, \mathrm{e}^{\mathrm{j}\omega_c}) = \frac{1}{M_{\mathrm{r}}M_{\mathrm{c}}}\sum_{k_{\mathrm{r}}=0}^{M_{\mathrm{r}}-1}\sum_{k_{\mathrm{c}}=0}^{M_{\mathrm{c}}-1}X(\mathrm{e}^{\mathrm{j}(\omega_r-2k_{\mathrm{r}}\pi)/M_{\mathrm{r}}}, \mathrm{e}^{\mathrm{j}(\omega_c-2k_{\mathrm{c}}\pi)/M_{\mathrm{c}}})
$$

式（7-16）表明，二维抽取后的频谱等于原信号频谱分别在行列方向上膨胀 M_{r} 和 M_{c} 倍后再分别平移 $2k_{\mathrm{r}}\pi$ 和 $2k_{\mathrm{c}}\pi$ 得到的 $M_{\mathrm{r}}M_{\mathrm{c}}$ 个谱之和，幅度则为原来的 $1/M_{\mathrm{r}}M_{\mathrm{c}}$。$(M_{\mathrm{r}}, M_{\mathrm{c}})=(2, 2)$ 时抽取后谱关系如图 7-13 所示。从图中可以看出，因为 $\boldsymbol{X}$ 的截止频率 $\omega_{\mathrm{r}}(\omega_{\mathrm{c}})>\pi/2$，所以 $\boldsymbol{Y}_{\mathrm{D}}$ 发生了重叠。

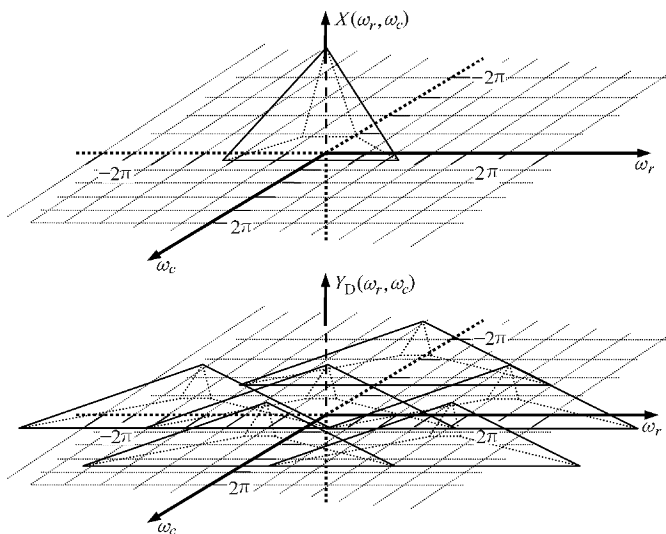


图 7-13　$(M_{\mathrm{r}}, M_{\mathrm{c}})=(2, 2)$ 二维抽取器频谱特性

由图 7-13 还可看出，行频的截止频率 $\omega_r>\pi/2$，而列频截止频率 $\omega_c<\pi/2$，所以分别进行 2∶1 下取样后 $\boldsymbol{Y}_{\mathrm{D}}$ 在行方向上有重叠而列方向是分开的。同一维抽取和内插处理相同，为消除频谱混叠及正确还原，二维信号也有相应的抽取和内插滤波器。

利用 MATLAB 实现 $(M_{\mathrm{r}}, M_{\mathrm{c}})=(3, 2)$ 的抽取和 $(L_{\mathrm{r}}, L_{\mathrm{c}})=(3, 2)$ 的内插过程及谱分析结果如图 7-14 所示。抽取后的谱 Y_{D} 发生了膨胀，内插后的谱 Y_{E} 则产生了镜像。

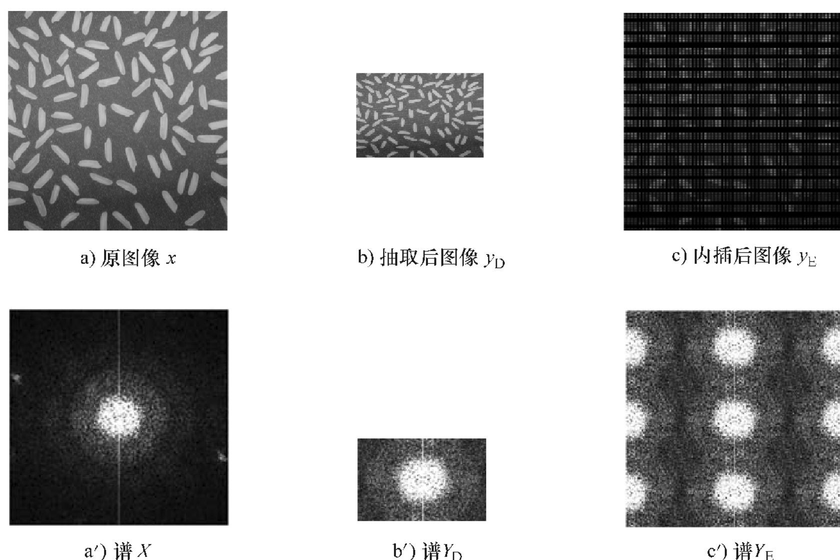


图 7-14　二维信号抽取和内插实验

相应的利 MATLAB 实现的代码如下：

```
function specDI2(N)
if nargin = =0
   N =128;M = [3,2];L = [3,2];
end
clc,clf,close all;
xrd = imread('rice.png');
ifndims(xrd) = =3
   xrd = xrd(:,:,1);
end
[r,c] = size(xrd);nr = nextpow2(r);nc = nextpow2(c);
rr = pow2(nr);cc = pow2(nc);
if rr ~ = r
   r = pow2(nr -1);
end
if cc ~ = c
   c = pow2(nc -1);
end
x = xrd(1:r,1:c);
dx = decS(x,M);getSP(x,dx,N);
ix = intS(dx,L);getSP(dx,ix,N);

function dx = decS(x,M)
```

```
n=size(x);N=fix(n./M);
dx=x(1:M(1):N(1)*M(1),1:M(2):N(2)*M(2));
imshow(x),figure,imshow(dx);

function ix=intS(x,L)
n=size(x);N=L.*n;ix=uint8(zeros(N(1),N(2)));
ix(1:L(1):N(1)-L(1)+1,1:L(2):N(2)-L(2)+1  )=x;
figure,subplot(1,2,1),subimage(x),axis off;
subplot(1,2,2),subimage(ix),axis off;

function p=getSP(sig,dx,N)
[r,c]=size(sig);
xsp=fftshift(fft2(sig,r,c));xA=abs(xsp);
X=(xA-min(min(xA)))./(max(max(xA))-min(min(xA)))*256;
[r,c]=size(dx);
dxsp=fftshift(fft2(dx,r,c));dxA=abs(dxsp);
YD=(dxA-min(min(dxA)))./(max(max(dxA))-min(min(dxA)))*256;
figure,subplot(1,2,1),imshow(X);subplot(1,2,2),imshow(YD);
```

7.2 滤波器组

7.2.1 双通道正交镜像滤波器组（2QMFB）

数字滤波器组是多个滤波器的组合，这些滤波器具有共同的输入或者输出。图 7-15a 为分析滤波器组，$H_k(z)$ 是分析滤波器，$y_k(n)$ 是子带信号，图中输入信号 $x(n)$ 被分解成 N 个子带信号。图 7-15b 为综合滤波器组，$F_k(z)$ 是综合滤波器，子带信号 $\hat{y}_k(n)$ 被综合成输出信号 $\hat{x}(n)$。

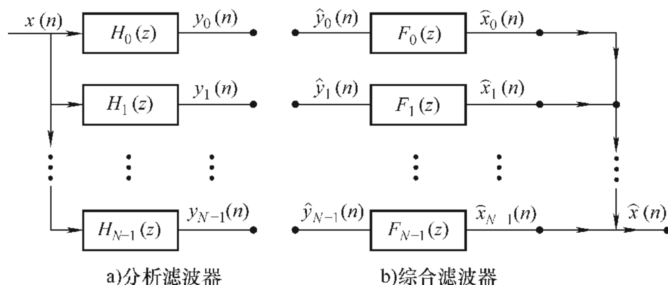


图 7-15　数字滤波器组

输入信号 $x(n)$ 进入 N 个通道，每通道中有一个分析滤波器 $H_k(n)$，$k=0$，1，…，$N-1$。设是 $x(n)$ 宽频带信号，经过各通道中的带通滤波器后被分成 N

个子频带信号 $y_k(n)$。显然，$y_k(n)$ 是窄带信号，如果分析滤波器组中第 k 个通道滤波器的脉冲响应 $h_k(n)$ 和转移特性 $H_k(z)$ 具有如下关系：

$$\begin{aligned} h_k(n) &= h_0(n)\mathrm{e}^{\mathrm{j}\frac{2\pi}{N}kn} \\ H_k(z) &= H_0(zW^k) \end{aligned} \quad k=0,1,\cdots N-1; W=\mathrm{e}^{-\mathrm{j}\frac{2\pi}{N}} \tag{7-17}$$

则称这个滤波器组为均匀 DFT 滤波器组。其中各个滤波器幅频特性如图 7-16 所示。滤波后各通道信号 $y_k(n)$ 是窄带信号，因此其抽样率可以降低。如果 $x(n)$ 是满带信号即 $X(\mathrm{e}^{\mathrm{j}\omega})$ 的频谱占满 $-\pi$ 到 $+\pi$ 的区域，而各个通道的信号都具有相同的带宽 B。各子带带宽 $B=2\pi/N$，则抽样率最多可以降到原来的 $1/N$，若低于 $1/N$，则会出现混叠。即通过各通道滤波器后的信号可进行抽取因子为 D 等于或小于 K 的抽取。在综合滤波器组中输入信号 $\hat{y}_k(n)$，$k=0$，1，…，$N-1$ 先进行零值内插后经过综合滤波器 $F_k(j\omega)$ 即保留了所需要的子频带，得到相应的子带信号 $\hat{x}_k(n)$，将所有子带信号相加就得到综合信号 $\hat{x}(n)$，即：

$$\begin{aligned} \widehat{x}(n) &= \sum_{k=0}^{N-1}\widehat{x}_k(n) \\ \widehat{X}(\mathrm{e}^{\mathrm{j}\omega}) &= \sum_{k=0}^{N-1}\widehat{X}_k(\mathrm{e}^{\mathrm{j}\omega}) \end{aligned} \tag{7-18}$$

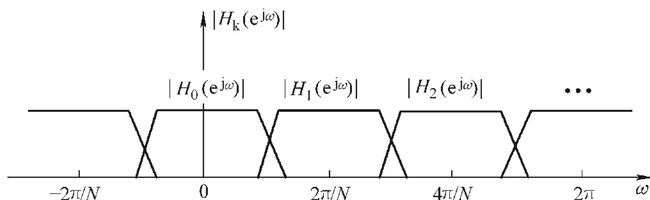


图 7-16　均匀 DFT 滤波器组中各滤波器的幅频特性

图 7-17 是 M 子带变换型滤波器组，将信号均匀分割成 M 个子带，为 M 带最大抽取均匀滤波器组。

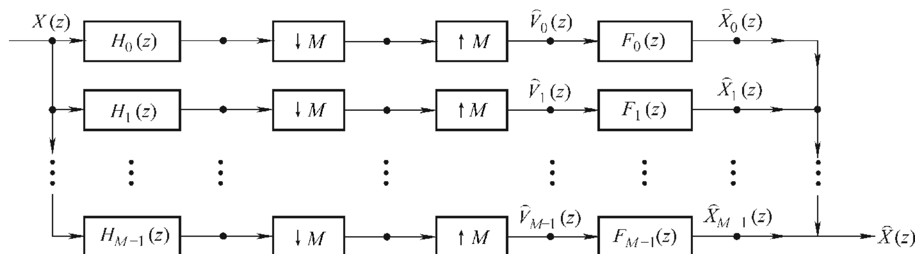


图 7-17　M 通道滤波器组

图 7-17 中各通道子带信号通过内插器和综合滤波器重建信号，其中内插器用于恢复采样率，综合滤波器用于消除镜像谱。图 7-17 结构又称为正交镜像滤波器组。信号经过分析滤波器组后成为多个频带变窄的子带，为了减少数据量，对各子带信号进行降采样即抽取过程。相应的在恢复信号时需要升采样即零插过程。由于滤波器的非理想特性，抽样率的变换导致了混叠的产生。结合式（7-3）和式（7-8），图 7-17M 通道 QMFB 的输出等于：

$$\begin{aligned}\hat{X}(z) &= \sum_{k=0}^{M-1}\hat{X}_k(z) = \frac{1}{M}\sum_{m=0}^{M-1}X(zW^m)\sum_{k=0}^{M-1}H_k(zW^m)F_k(z)\\ &= \sum_{m=0}^{M-1}X(zW^m)A_m(z) = X(z)A_0(z) + \sum_{m=1}^{M-1}X(zW^m)A_m(z)\end{aligned} \tag{7-19}$$

式中

$$A_m(z) \overset{def}{=} \frac{1}{M}\sum_{m=0}^{M-1}H_m(zW^m)F_m(z) \tag{7-20}$$

且可看出，$X(zW^m)$ $m=1, 2, \cdots, M-1$ 是输入信号 $X(z)$ 在频域移位 $2m/M$ 后的结果，因此它是混叠分量。为消除其影响而重构原信号，应使 $A_m(z)$ 为 $0(m\neq 0)$。这样，式（7-19）就变成式（7-21），从而在输出信号中消除了混叠成分。

$$\hat{X}(z) = X(z)A_0(z) = \frac{1}{M}\sum_{k=0}^{M-1}H_k(z)F_k(z)X(z) \tag{7-21}$$

二通道（$M=2$）QMFB 是滤波器组的一个特例，其子带编解码系统框图如图 7-18所示。

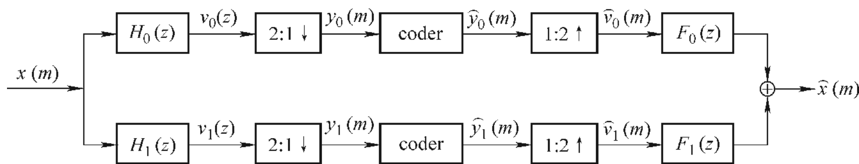


图 7-18　一维 2 子带 QMFB

由图 7-18 可知，信号经解析滤波器可得：

$$\begin{cases}V_0(z) = H_0(z)X(z)\\ V_1(z) = H_1(z)X(z)\end{cases} \tag{7-22}$$

经 2∶1 下取样得：

$$\begin{cases}Y_0(z) = \dfrac{1}{2}[H_0(z^{1/2})X(z^{1/2}) + H_0(z^{1/2}W)X(z^{1/2}W)]\\ Y_1(z) = \dfrac{1}{2}[H_1(z^{1/2})X(z^{1/2}) + H_1(z^{1/2}W)X(z^{1/2}W)]\end{cases} \tag{7-23}$$

其中，$W=e^{j2\pi/2}=-1$。经综合滤波器得：

$$\widehat{X}(z)=F_0(z)\widehat{V}_0(z)+F_1(z)\widehat{V}_1(z)=F_0(z)\widehat{Y}_0(z^2)+F_1(z)\widehat{Y}_1(z^2) \tag{7-24}$$

若忽略编码误差，将式（7-23）代入到式（7-24）中并整理得：

$$\widehat{X}(z)=\frac{1}{2}[F_0(z)H_0(z)+F(z)_1H_1(z)]X(z)+\frac{1}{2}[F_0(z)H_0(zW)+F_1(z)H_1(zW)]X(zW) \tag{7-25}$$

式（7-25）的第二项为混叠部分，若综合滤波器如下选取：

$$\begin{cases} F_0(z)=2H_1(zW) \\ F_1(z)=-2H_0(zW) \end{cases} \tag{7-26}$$

则混叠分量为零，系统可无失真还原信号，其传输函数如下：

$$\begin{aligned}\widehat{X}(z)&=[H_0(z)H_1(zW)-H_0(zW)H_1(z)]X(z)\\ \Rightarrow T(z)&=H_0(z)H_1(zW)-H_0(zW)H_1(z)\end{aligned} \tag{7-27}$$

若 $T(z)$ 是具有线性相位的全通网络即 $T(z)=cz^{-k}$（c，k 是常数），则输出与输入关系为

$$\widehat{X}(z)=cz^{-k}X(z)\Rightarrow\widehat{x}(n)=cx(n-n_0) \tag{7-28}$$

7.2.2　2QMFB 完全重建系统

如果分析滤波器组满足 $H_1(z)=H_0(-z)$，对应的综合滤波器组 $F_0(z)=2H_0(z), F_1(z)=-2H_1(z)$，即只要确定低通滤波器 $H_0(z)$，系统所有其他的滤波器特定都确定下来。系统传输函数如下：

$$T(z)=H_0^2(z)-H_1^2(z)=H_0^2(z)-H_0^2(-z) \tag{7-29}$$

对分析滤波器组和综合滤波器组进行 I 型多相分解（参见附录 B）得：

$$\begin{cases} H_0(z)=E_0(z^2)+z^{-1}E_1(z^2) \\ H_1(z)=E_0(z^2)-z^{-1}E_1(z^2) \end{cases} \qquad \begin{cases} F_0(z)=E_0(z^2)+z^{-1}E_1(z^2) \\ F_1(z)=-E_0(z^2)+z^{-1}E_1(z^2) \end{cases} \tag{7-30}$$

将式（7-30）代入到式（7-29）得到传输函数为

$$T(z)=4z^{-1}E_0(z^2)E_1(z^2) \tag{7-31}$$

显然，只有 $E_0(z)$ 和 $E_1(z)$ 分别为延迟因子，即 $E_0(z)=c_0z^{-a}$，$E_1(z)=c_1z^{-b}$ 时（a，b 是常数），系统才能完全消除幅度失真。这样，分析滤波器 $H_0(z)=c_0z^{-2a}+c_1z^{-2b-1}$，显然此滤波器不具备低通特性。实际中，往往采用功率对称的 FIR 来设计 2QMFB。

若 FIR 滤波器传输函数 $H(z)$ 满足如下条件：

$$\tilde{H}(z)H(z)+\tilde{H}(-z)H(-z)=c\,(c>0) \tag{7-32}$$

则称 $H(z)$ 是功率对称滤波器，式（7-32）中 $\tilde{H}(z)$ 是 $H(z^{-1})$ 的共轭。在

2QMFB 中，如果 $H_0(z)$ 是功率对称滤波器，$H_1(z)$ 作如下选取：

$$H_1(z) = -z^{-(N-1)}\tilde{H}_0(-z) \tag{7-33}$$

式中 N 是奇数。根据式（7-27）可知系统传输特性为 $cz^{-(N-1)}$，系统是完全重建的！则滤波器组的时域表达式如下：

$$h_1(n) = (-1)^{n+1}h_0^*(N-1-n) \quad f_0(n) = 2h_0^*(N-1-n)$$
$$f_1(n) = 2(-1)^n h_0^*(n) \tag{7-34}$$

综合以上，有以下结论：

1）$|F_k(\mathrm{e}^{\mathrm{j}\omega})| = |H_k(\mathrm{e}^{\mathrm{j}\omega})|$，即分析滤波器组与综合滤波器组具有相同的幅频特性。

2）$|H_1(\mathrm{e}^{\mathrm{j}\omega})| = |H_0(-\mathrm{e}^{-\mathrm{j}\omega})|$，即若 $H_0(\omega)$ 是低通滤波器，则 $H_1(\omega)$ 是高通滤波器。

3）$|H_0(\mathrm{z})|^2 + |H_1(\mathrm{z})|^2 = c$，即分析滤波器组满足功率互补特性。

4）$\tilde{H}(z)H(z)$是半带滤波器（参见附录 C）。

7.2.3　全相位半带滤波器设计

由上面结论可知，如果 $h(n)$ 是实序列，则半带滤波器 $G(\mathrm{e}^{\mathrm{j}\omega})$ 满足：

$$G(\mathrm{e}^{\mathrm{j}\omega}) = H(\mathrm{e}^{\mathrm{j}\omega})H^*(\mathrm{e}^{\mathrm{j}\omega}) = |H(\mathrm{e}^{\mathrm{j}\omega})|^2 \geqslant 0 \tag{7-35}$$

即实系数单位冲激响应设计的半带滤波器幅频特性横为非负。这样，只要设计出半带滤波器 $G(z)$，然后再谱分解即可得到功率对称滤波器 $H_0(z)$，继而得到 $H_1(z)$，而$H_0(z)$ 和 $H_1(z)$ 满足功率互补特性，即按照（7-35）设计就可以得到 PR 的内插滤波器 $F_0(z)$ 和 $F_1(z)$。

目前，半带滤波器常用的设计方法有窗函数法、拉格朗日插值法和单带滤波器法，下面分别给以简单介绍。

1. 窗函数法

假设要设计的半带滤波器截至频率 $\omega_c = \pi/2$，令理想滤波器的$|H_\mathrm{d}(\mathrm{e}^{\mathrm{j}\omega})| = 1$，对于$|\omega| \leqslant \pi/2$，其余为零，则

$$h_\mathrm{d}(n) = \frac{1}{2\pi}\int_{-\pi/2}^{\pi/2}\mathrm{e}^{\mathrm{j}n\omega}\mathrm{d}\omega = \frac{\sin(n\pi/2)}{n\pi} \tag{7-36}$$

显然，单位冲激响应 $h_\mathrm{d}(n)$ 偶数项为 0 且中心点等于 0.5，偶对称。所以，它是零相位的半带滤波器，选择长度为 $N = 4J-1$ 的窗函数 $w(n)$，并与移位 $2j-1$ 后的 $h_\mathrm{d}(n)$ 相乘即得到最终半带滤波器的单位冲激响应 $h(n)$ 为

$$h(n) = h_\mathrm{d}[n-(2j-1)]w(n) \tag{7-37}$$

2. 拉格朗日插值法

首先按照下式求出半带滤波器的单位冲激响应为

$$h(2n-1)=\frac{(-1)^{n+J-1}\prod_{k=1}^{2J}(J-k+0.5)}{(J-n)!(J-1+n)!(2n-1)}\quad n=1,2,\cdots,J \tag{7-38}$$

$h(n)$ 的总长度为 $N=4J-1$。则对应的转移函数如下：

$$H(z)=0.5+\sum_{n=1}^{J}h(2n-1)(z^{-2n+1}+z^{2n-1}) \tag{7-39}$$

3. 单带滤波器法

首先用 Chebyshew 最佳一致逼近法设计出一个单带滤波器 $G(z)$。所谓“单带”是令 $G(z)$ 的通带频率为 $2\omega_p$，阻带频率为 π，即从 $2\omega_p$ 到 π 是过渡带，因此，$G(z)$ 没有阻带。令 $g(n)$ 的长度为 $(N+1)/2$，因为半带滤波器长度为 $N=4J-1$，所以 $g(n)$ 的长度为 $2J$。$G(z)$ 和 $g(n)$ 求出后，再对 $g(n)$ 做 2 倍插值，并令插值后的序列中心等于 0.5，即

$$H(z)=\frac{1}{2}[G(z^2)+z^{-(N-1)/2}] \tag{7-40}$$

这样，$H(z)$ 是半带滤波器，其截至频率变为 ω_p，通带和阻带内的波纹数分别是 $G(z)$ 的一半。

由第 2 章中的结论可知，ap^1 的频率取样序列 $\boldsymbol{H}$ 反映了系统传输特性，利用 ap^1 对 $\boldsymbol{H}$ 的线性性，可方便地实现严格子带互补的半带滤波器组。为满足在 $\omega_m=\pi/2$ 时的幅度等于 0.5，只需设置 $\boldsymbol{H}$ 中下标为 $(\omega_m/(2\pi/N))=N/4$ 处的值为 0.5，同时考虑到线性相位，对 ap^1 的频率取样序列 $\boldsymbol{H}$ 可作如下设定：

1）$\boldsymbol{H}$ 中下标为 0 到 $N/4-1$ 及 $3N/4$ 到 $N-1$ 的元素等于 1。

2）$\boldsymbol{H}$ 中下标 $N/4$ 及 $3N/4-1$ 的元素等于 0.5。

3）$\boldsymbol{H}$ 中其余元素等于 0。

考虑如下低高通序列：

$$\begin{aligned}\boldsymbol{H}_0&=[1\ \underbrace{1\ \cdots\ 1}_{N/4\text{个}}\ 0.5\ 0\ \cdots 0\ \cdots\ 0\ 0.5\ \underbrace{1\ \cdots\ 1}_{N/4\text{个}}]\\ \boldsymbol{H}_1&=[0\ 0\ \cdots\ 0\ 0.5\ \overbrace{1\ \cdots 1\ \cdots\ 1}^{N/2-1\text{个}}\ 0.5\ 0\ \cdots\ 0]\end{aligned} \tag{7-41}$$

低通序列的单位脉冲响应序列等于：

$$\begin{aligned}\boldsymbol{h}_0&=\frac{1}{N}\sum_{k=0}^{N-1}H_0(k)W_N^{-kn}=\frac{1}{N}\left[1+2\sum_{k=1}^{N/4-1}\cos\left(\frac{2k\pi}{N}n\right)+\cos\left(\frac{n\pi}{2}\right)\right]\\ &=\frac{1}{N}\frac{\sin(n\pi/2)}{\sin(2n\pi/N)}[1+\cos(2n\pi/N)]\\ &=\frac{1}{N}\sin(n\pi/2)\cot(n\pi/N)\end{aligned} \tag{7-42}$$

又 $\boldsymbol{H}_0+\boldsymbol{H}_1=1$，所以由式（7-42）可得高通序列的单位脉冲响应序列如下：

$$\boldsymbol{h}_1 = \frac{1}{N}\sum_{k=0}^{N-1}[1-H_0(k)]W_N^{-kn} = \delta(n) - \boldsymbol{h}_0 \tag{7-43}$$

由式（7-42）和式（7-43）易知，$h_1(0)=h_0(0)=0.5$，其余偶数项均等于 0 而奇数项反相。结合式（2-51）有

$$\begin{aligned} H_0(\omega) &= 0.5C(0) + \frac{2}{N}\sum_{n=1}^{N-1}C(n)\sin(n\pi/2)\cot(n\pi/N)\cos(n\omega) \\ H_1(\omega) &= 0.5C(0) - \frac{2}{N}\sum_{n=1}^{N-1}C(n)\sin(n\pi/2)\cot(n\pi/N)\cos(n\omega) \end{aligned} \tag{7-44}$$

归一化卷积窗 $C(0)=1$。由式（7-44）易知下式成立：

$$\begin{cases} H_0(\omega) + H_1(\omega) = 1 \\ H_1(\omega) = H_0(\omega+\pi) \end{cases} \tag{7-45}$$

由上面过程可知，按照式（7-41）设置长度为 N 的 $\boldsymbol{H}$ 对应的 ap^1 是长度为 $2N-1$ 的半带滤波器。半带滤波器 $h(n)$ 满足：

$$\begin{cases} h(n) = h(N-2-n) & 0\leqslant n\leqslant N-2 \\ h(n) = h(3N-2-n) & N-1\leqslant n\leqslant 2N-2 \end{cases} \tag{7-46}$$

ap^1 半带滤波器法与传统 3 种方法生成的滤波器的 MATLAB 代码如下：

```
function semiB(N)

if nargin = =0
    J =4;N =4* J -1;delta =0.05;
funame ={'winSB(N,J)','lgSB(N,J)','sbSB(N,J)','apSB(N,J)'};
end
clc,close all
for id =1:4
    hn =eval(funame{id});[HW,w] =toGetHW(hn);
    toDraw(hn,w,HW,id,delta);
end

function hn =winSB(N,J)
n =[0:N -1] -2* J +1;n(n = =0) =eps;
hd =sin(n* pi/2)./(n* pi);hn =hd.* blackman(N)';

function hn =lgSB(N,J)
h =zeros(1,2* J -1);
for i =1:J
    h(2* i -1) =( -1)^(i +J -1)* prod(J +0.5 -[1:2* J])...
        /factorial(J -i)/factorial(J -1 +i)/(2* i -1);
end
hn =[fliplr(h),0.5,h];
```

```
function hn = sbSB(N,J)
wp =0.33* pi* 2;wps =0:0.01* pi:wp;ws =pi;
f = [wps/pi,ws/pi];a = [ones(size(wps)),zeros(size(ws))];
h = remez(2* J -1,f,a);
hn = zeros(1,N);hn(1:2:N) =h/2;hn(2* J) =0.5;

function hn = apSB(N,J)
NN =2* J;H = zeros(1,NN);H(1:NN/4) =1;H(NN/4 +1) =0.5;
H(NN* 3/4 +2:N) =1;H(NN* 3/4 +1) =0.5;
h = ifft(H,NN);win = conv(triang(NN),triang(NN));win =win/max(win);
hn = [fliplr(h(2:end))h]. * win';

function[HW,w] = toGetHW(hn)
N = length(hn);J = (N +1)/4;

w =0:2* pi/N^2:pi;cwn =2* cos(w'* [1:2* J -1]);
hns = repmat(hn((N +3)/2:end),length(w),1);
HW = sum(hns. * cwn,2) +hn((N +1)/2);

function p = toDraw(hn,w,HW,id,delta)
tle = {'Window','Lagrange','semiband','all -phase'};err =10^(-2);
if 3 = =id
   figure
end
subplot(2,2,mod(id -1,2)* 2 +1),set(gca,'FontSize',16);
stem(hn,'k - -','fill','MarkerSize',6);
gridon;set(gca,'XLim',[1 length(hn)]);
xlabel('\itn');ylabel('\ith\rm(\itn\rm)');title(tle{id});
subplot(2,2,mod(id -1,2)* 2 +2),set(gca,'FontSize',16);
plot(w,HW,'k','LineWidth',3);grid on;set(gca,'YLim',[ -0.3 1.3]);
xlabel('\itn');ylabel('\itH\rm(\it\rmj\it\omega\rm)');title(tle{id});
x2 =w(abs(HW -0.5) <err);hold on;
drawDot(x2,0.5,1);

pdelta =abs(HW -1 +delta) <err;xdelta =abs(HW - delta) <err;
xp =w(pdelta);xp =mean(xp);HWp =mean(HW(pdelta));
drawDot(xp,HWp,2);
xs =w(xdelta);xs =mean(xs);HWs =mean(HW(xdelta));
drawDot(xs,  HWs,3);
pps =round(100* (xp +xs)/pi)/100;
strpps =strcat(num2str(pps),'\pi');
```

```
expr='\it \omega_p \rm + \it \omega_s \rm=';
h=text(3,0.8,strcat(expr,strpps));set(h,'FontSize',16);

functiondrawDot(x,y,id)
xx=round(x* 100/pi)/100;
dotname={strcat(num2str(xx),'\pi'),'\it \omega_p','\it \omega_s'...
    num2str(y),'1 - \it \delta','\it \delta'};
plot(x,y,'ko','MarkerSize',6,'MarkerFaceColor','k');
xr=0:0.01:x;yr=0:0.01:y;
h=plot(xr,ones(size(xr))* y,'k--',x* ones(size(yr)),yr,'k--');
set(h,'LineWidth',2);
h1=text(x-0.2,-0.1,dotname{id});
h2=text(x+0.05,y+0.05,dotname{id+3});
set([h1,h2],'FontSize',16);
```

运行结果如图 7-19 所示。由图 7-19 对比可看出，半带方法得到的滤波器具

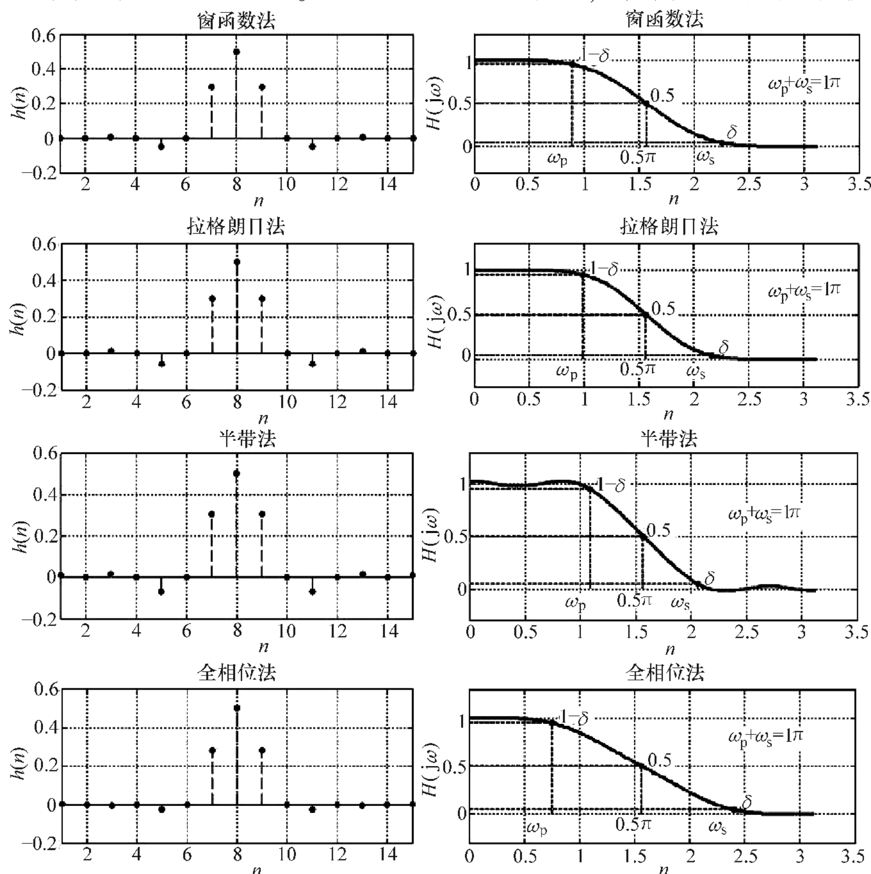


图 7-19　全相位法与其他 3 种设计半带滤波器对比

有最小过渡带宽度，但通带和阻带内的波纹也最多且出现负幅值，因此需要先进行抬升后才可以进行谱分解来得到分析滤波器 $\boldsymbol{H}_0$；ap 法同窗函数法和拉格朗日法在通带和阻带保持很好的平滑性，但不同窗函数也会出现负幅值现象，拉格朗日法又相对复杂；ap[1nd]法设计得到的半带滤波器幅值则横为非负，且设计方法简单，物理意义明确。

由第 2 章结论可知，具有线性相位的 ap 传输特性均可表示为余弦波的加权线性组合。观察式（7-44），令：

$$a(n)=\frac{2C(n)\sin(n\pi/2)\cot(n\pi/N)}{N} \tag{7-47}$$

对于任意的角频率 $\theta(-\pi/2\leqslant\theta\leqslant\pi/2)$ 结合 $a(n)$ 偶数项为零，当 $n=2k-1(k\in Z)$ 时，有

$$\begin{aligned}a(n)\cos\left[n\left(\theta+\frac{\pi}{2}\right)\right]&=a(n)\cos\left(n\theta+k\pi-\frac{\pi}{2}\right)\\&=-a(n)\cos\left(n\theta-k\pi+\frac{\pi}{2}\right)\\&=a(n)\cos\left[n\left(\frac{\pi}{2}-\theta\right)\right]\end{aligned} \tag{7-48}$$

即 ap 曲线在角频率 π/2 处满足奇对称，如图 7-20 所示。

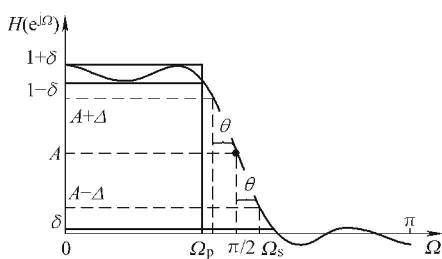


图 7-20　全相位半带滤波器幅频关于 π/2 奇对称

所以，如果冲激响应序列满足 3 个条件中的第 2 个即 $N=4L$（L 是整数），这样就可以通过设置 $\boldsymbol{H}$ 使在角频率 π/2 处等于 0.5；若 N 不是 4 的整数倍即无法直接设置角频率 π/2 处的值，但通过式（7-48）可知，设置 $\boldsymbol{H}$ 在角频率 π/2 左右点分别为 1 和 0（低通），则 ap 法得到的滤波器仍满足 3 个条件中的第二个，即

$$2\cdot\frac{\pi}{2}=\frac{2\pi}{N}(L-1+L)\Rightarrow N=4L-2 \tag{7-49}$$

结合式（7-41），这时 H_0结构如下：

$$\boldsymbol{H}_0=[1\ \underbrace{1\ \cdots\ 1}_{(N-2)/4\text{个}}\ \underbrace{0\ 0\ \cdots 0\ \cdots\ 0\ 0}_{N/2\text{个}}\ \underbrace{1\ \cdots\ 1}_{(N-2)/4\text{个}}] \tag{7-50}$$

按照式（7-42）推导的过程，低通滤波器冲激响应为

$$\boldsymbol{h}_0=\frac{1}{N}\sum_{k=0}^{N-1}H_0(k)W_N^{-kn}=\frac{1}{N}\left[1+2\sum_{k=1}^{N/4-1}\cos\left(\frac{2k\pi}{N}n\right)\right]=\frac{1}{N}\frac{\sin(n\pi/2)}{\sin(n\pi/N)}\quad(7\text{-}51)$$

由式（7-51）易知，$h_0(0)=0.5$ 且偶数项为 0 并满足式（7-47）对称性。结合式（2-51）得到：

$$H_0(\omega)=0.5C(0)+\frac{2}{N}\sum_{n=1}^{N-1}\frac{C(n)\sin(n\pi/2)}{\sin(n\pi/N)}\cos(n\omega)\quad(7\text{-}52)$$

因为 $C(0)=1$，所以 $H_0(\pi/2)=0.5$，即 $\boldsymbol{H}_0$ 也是半带滤波器。

结合式（7-41）和式（7-42），假定低频取样序列 $\boldsymbol{H}_0$ 作如下定义：

$$\boldsymbol{H}_0=[1\ \underbrace{1\ \cdots\ 1\ \alpha}_{N/4\text{个}}\ \underbrace{0\ \cdots 0\ \cdots\ 0}_{N/2-1\text{个}}\ \underbrace{\beta\ 1\ \cdots\ 1}_{N/4\text{个}}]\quad(7\text{-}53)$$

则容易求出：

$$\boldsymbol{h}_0=\frac{1}{N}\left[1+2\sum_{k=1}^{N/4-1}\cos\left(\frac{2k\pi}{N}n\right)+(\alpha+\beta)\cos\left(\frac{n\pi}{2}\right)+\mathrm{j}(\alpha-\beta)\sin\left(\frac{n\pi}{2}\right)\right]\quad(7\text{-}54)$$

若 α 与 β 不等，$\boldsymbol{H}_0$ 不再满足线性相位条件，因此 $\boldsymbol{h}_0$ 出现复数部分。但是，只要 $\alpha+\beta=1$，则容易得出下面结果：

$$h_0(2n)=\begin{cases}0, & n\neq 0\\ 0.5, & n=0\end{cases}\quad(7\text{-}55)$$

所以，基于 $\boldsymbol{h}_0$ 构建的 ap 就是半带滤波器。取 $\alpha=1$（$\beta=1$ 时可类似处理），半带滤波器的传输函数如下：

$$H_0(\omega)=0.5C(0)+\frac{2}{N}\sum_{n=1}^{N-1}C(n)\sin(n\pi/2)\cot(n\pi/N)\cos(n\omega)+\frac{2}{N}\sum_{n=1}^{N-1}C(n)\sin(n\pi/2)\sin(n\omega)\quad(7\text{-}56)$$

在频率点 π/2 处，幅值等于：

$$\begin{aligned}H_0\left(\frac{\pi}{2}\right)&=0.5C(0)+\frac{2}{N}\sum_{n=1}^{N-1}C(n)\sin^2(n\pi/2)\\&=0.5+\frac{2}{N}\sum_{k=1}^{L}[C(n)+C(N-n)]\quad(7\text{-}57)\\&\begin{cases}=1 & \text{对于 ap}^{1\mathrm{ns}}\\ <1 & \text{对于 ap}^{1\mathrm{d}}\end{cases}\end{aligned}$$

之所以不再等于 0.5 是因为 $\boldsymbol{h}_0$ 不是实数。式（7-41）、式（7-50）和式（7-53）表示的条件分别称为条件 1、2 和 3。MATLAB 实现条件 2 和 3 的 $\text{ap}^{1\mathrm{nsd}}$ 半带滤波器代码（绘图部分省略，基本同函数 semiB）如下：

```
function apSB(N)
```

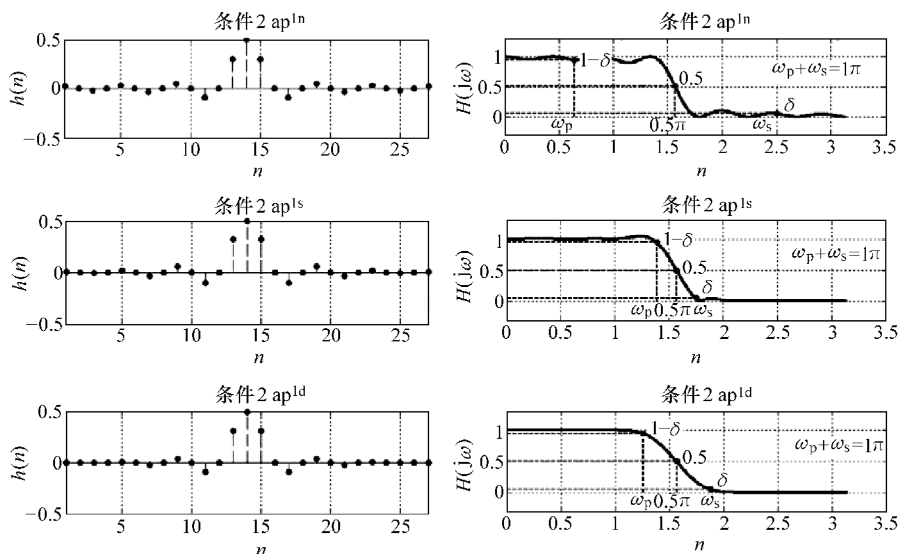
```
if nargin = =0
    L =4;delta =0.05;
end
clc,close all
for id =1:6
    hn =apSB23(L,id);[HW,w] =toGetHW(hn,id);
    toDraw(hn,w,HW,id,delta);
end

function hn =apSB23(L,id)
if id < 4
    N =4* L -2;H =zeros(1,N);H(1:L) =1;H(end -L +2 :end) =1;
else
    N =4* L;H =zeros(1,N);H(1:L) =1;H(end -L +1 :end) =1;
end
h =ifft(H,N);
switch id
    case { 1 4 }
        win =conv(boxcar(N),boxcar(N));
    case { 2 5 }
        win =conv(triang(N),boxcar(N));
    case { 3 6 }
        win =conv(triang(N),triang(N));
    otherwise
        error('bad parameters');
end
win =win/max(win);hn =[h(2:end),h].* win';
if id > 4
    disp(hn);
end

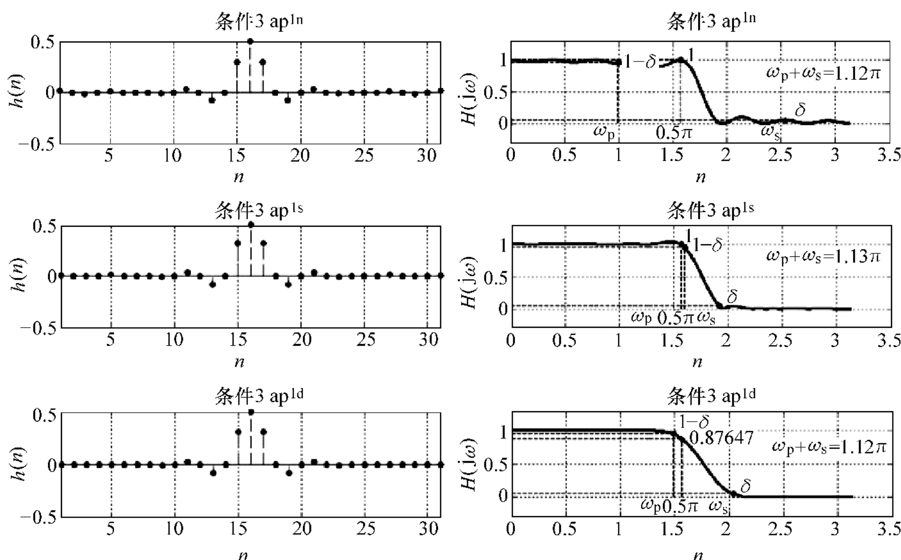
function [HW,w] =toGetHW(hn,id)
N =length(hn);w =0:2* pi/(N* 4^4): pi;
if id < 4
    cwn =2* cos(w'* [1 :(N -1)/2]);

else
    cwn =2* exp(-j* w'* [1 :(N -1)/2]);
end
hns =repmat(hn((N +3)/2:end),length(w),1);
HW =sum(real(hns.* cwn),2) +hn((N +1)/2);
```

运行结果如图 7-21 所示。可看出条件 2 下 ap^{1nsd}半带滤波器均通过了 π/2 频点，通带和阻带波纹幅度为 δ 的截止频率 $\omega_p+\omega_s=\pi$ 则证实在整个频带内关于 π/2 频点奇对称；在条件 3 下（$\alpha=1$ 时）ap^{1ns}半带滤波器在 π/2 频点幅值等于



a) 条件2下无窗、单窗和双窗ap半带滤波器特性



b) 条件3下无窗、单窗和双窗ap半带滤波器特性

图 7-21　满足条件 2、3 的 ap^{1nsd}半带滤波器

1，但对于 ap^{1d} 半带滤波器则为 0.87467，此结论在第 2 章中已证明过。$\omega_p+\omega_s>\pi$ 则说明在条件 3 下通带和阻带内的波动较条件 2 下减少。横向对比，ap^{1d} 的平滑性明显好于 ap^{1ns}，但牺牲了带宽。需要指出的是，条件 3 下 ap 半带滤波器 $h(n)$ 是复数，但满足中心共轭对称，因此具有式（7-56）的线性相位特性。图 7-21b 只画出了其实部。

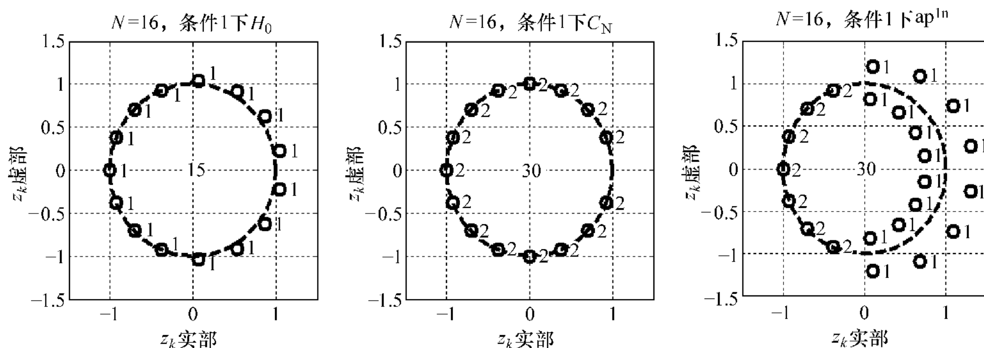
7.2.4　全相位半带滤波器的谱分解

长度为 N 的单边带 $\boldsymbol{H}_0$ 按上面 3 种方法设计生成的 ap 半带滤波器 $\boldsymbol{H}$ 的长度为 $2N-1$。对于条件 1 和 2，$h(n)$ 是实系数偶对称的，即 $h(n)=h(-n)$。假定 z_k 是 $H(z)$ 的零点，易知 z_k^{-1} 也是其零点；对于条件 1、2 和 3，$h(n)$ 均满足共轭偶对称，即 $h(n)=h^*(-n)$，所以，对于 $H(z)$ 的零点 z_k 有

$$H(z_k)=\sum_{n=-\infty}^{+\infty}h(n)z_k^{-n}=\sum_{n=-\infty}^{+\infty}h^*(-n)z_k^{-n}=\left[\sum_{n=-\infty}^{+\infty}h(n)\left[(z_k^*)^{-1}\right]^{-n}\right]^* \tag{7-58}$$
$$=H^*\left[(z^*)^{-1}\right]=0$$

即 z_k^* 和 $(z_k^*)^{-1}$ 也是 $H(z)$ 的零点。所以，对于条件 1、2 和 3 下的 ap 全相位滤波器零点以 z_k、z_k^{-1}、z_k^*、$(z_k^*)^{-1}$ 的双对形式出现；若 z_k 在单位圆上则以 z_k、z_k^* 的单对形式出现；若 z_k 等于 ±1 则重合为单点，若为其他实数，则以 z_k、z_k^{-1} 单对形式出现。

在满足条件 1、2、3 下，ap^{1nsd} 半带滤波器 $\boldsymbol{H}$、单边带 H_0 和卷积窗 $\boldsymbol{C}$ 的零点分布如图 7-22 所示（$L=4$，基窗为三角窗和矩形窗）。



a) 条件1下无窗ap $H_0/C/H$的零点分布

图 7-22　条件 1、2、3 下 ap^{1nsd} $\boldsymbol{H}_0/\boldsymbol{C}/\boldsymbol{H}$ 的零点分布

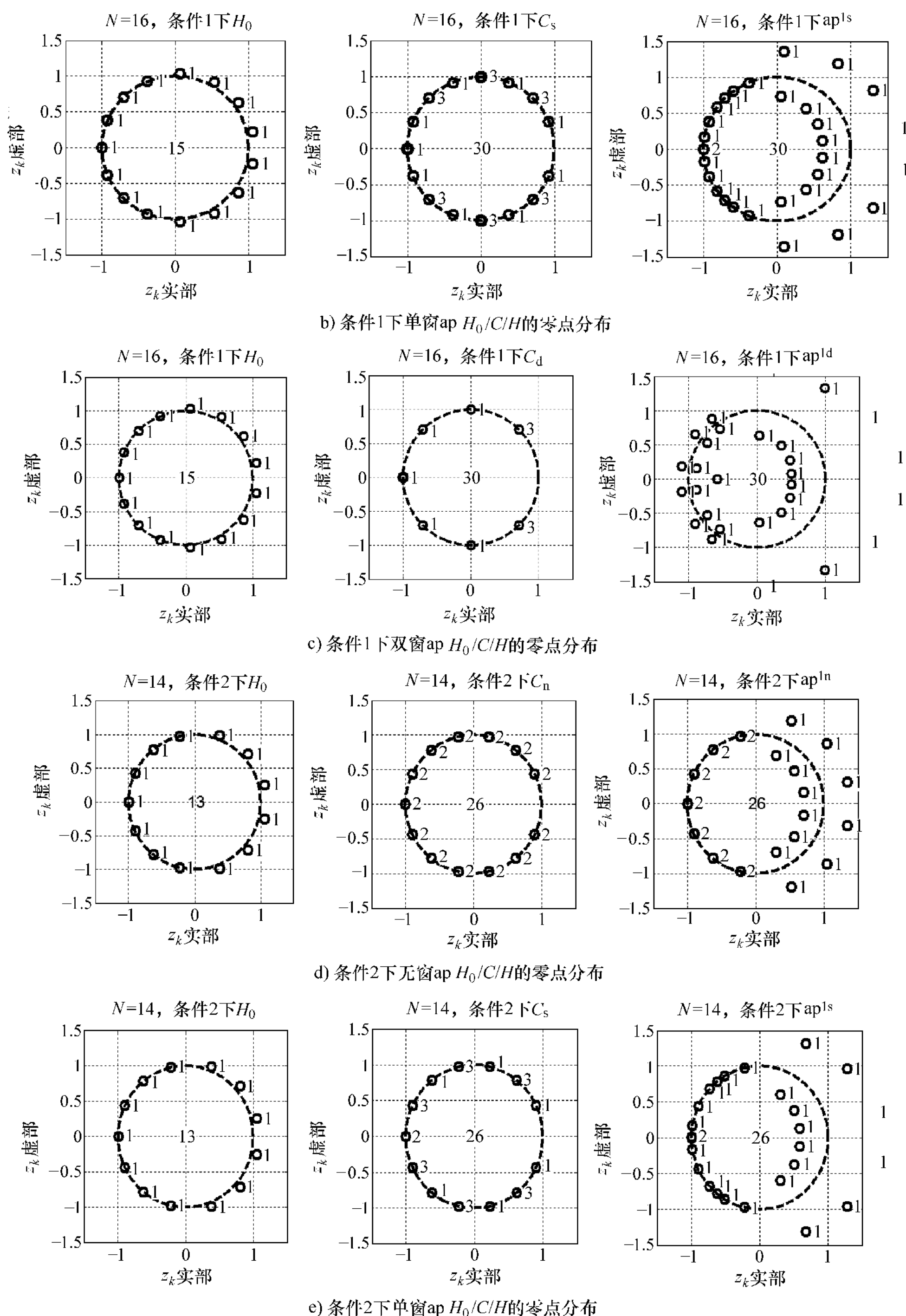
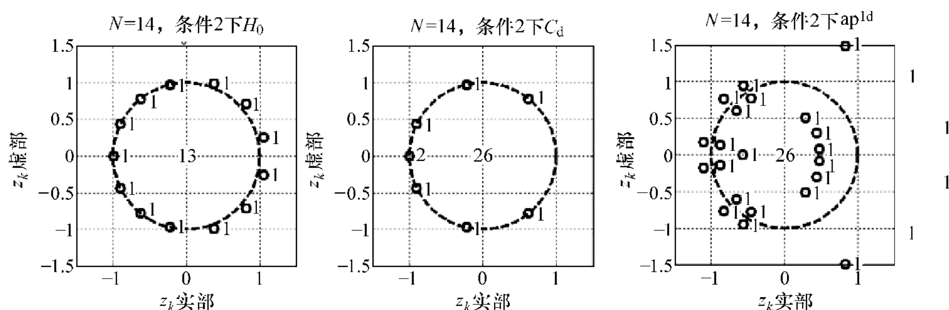
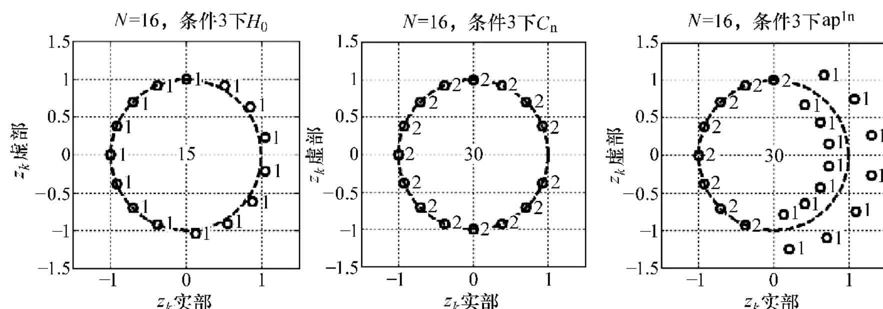


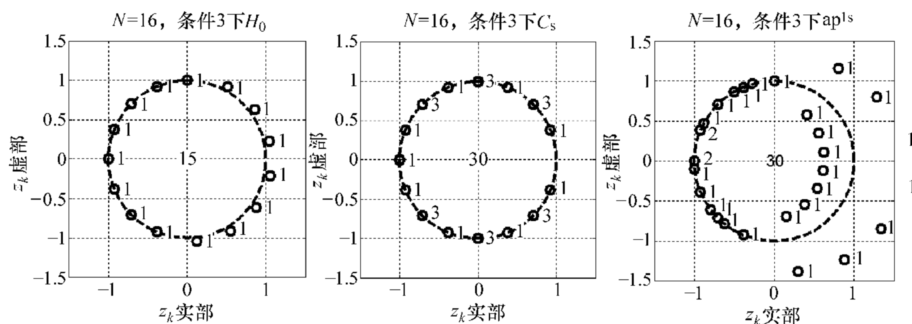
图 7-22　条件 1、2、3 下 $\text{ap}^{1\,\text{nsd}}$ $\boldsymbol{H}_0/\boldsymbol{C}/\boldsymbol{H}$ 的零点分布（续）



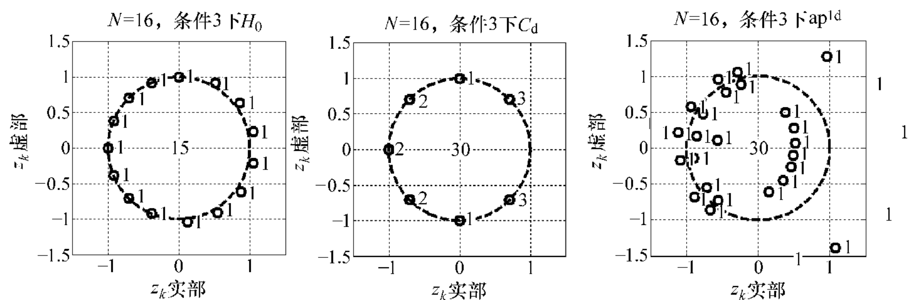
f) 条件2下双窗ap $H_0/C/H$的零点分布



g) 条件3下无窗ap $H_0/C/H$的零点分布



h) 条件3下单窗ap $H_0/C/H$的零点分布



i) 条件3下双窗ap $H_0/C/H$的零点分布

图 7-22　条件 1、2、3 下 $\mathrm{ap}^{1\,\mathrm{nsd}}$ $\boldsymbol{H}_0/\boldsymbol{C}/\boldsymbol{H}$ 的零点分布（续）

由图（7-22）实验结果可以得到如下结论：$\boldsymbol{H}_0$的单位圆上零点个数由$\boldsymbol{H}_0$零值个数决定，即条件1下等于$N/2-1$，条件2下等于$N/2$，条件3下等于$N/2$，且零点均匀分布；卷积窗$\boldsymbol{C}_\mathrm{n}$的零点均匀分布且是二重的，$\boldsymbol{C}_\mathrm{s}$的零点分布位置同$\boldsymbol{C}_\mathrm{n}$但不同位置的幂重数发生了改变，$\boldsymbol{C}_\mathrm{d}$的零点位置位于$\boldsymbol{C}_\mathrm{n}$的两相邻点中间，且幂重数改变的同时将新生靠得很近的零点。需要指出的是，采用的基窗类型不同，3种卷积窗的零点分布也会发生改变。$\mathrm{ap}^{1\mathrm{ns}}$半带滤波器的$\boldsymbol{H}$零点由单位圆上和内及对应的倒数（单位圆内和外）组成，而$\mathrm{ap}^{1\mathrm{d}}$半带滤波器的$\boldsymbol{H}$零点只由单位圆内及对应的倒数（单位圆外）组成。实现上述实验的 MATLAB 代码如下：

```
function zpD(L)
if nargin==0
    L=4;TN={'Cond 1','Cond 2','Cond 3'};
    apt={'\it\bfH\rm_0','\it\bfC_','ap^1^'};
end
clc,close all
for tp=1:3
    for id=1:3
        zeroP(L,tp,id,TN{tp},apt);
    end
end
function zeroP(L,tp,id,tn,apt)
switch tp
    case 1
        N=4*L;H=zeros(1,N);H(1:L)=1;H(L+1)=0.5;
        H(end-L+2:end)=1;H(end-L+1)=0.5;
    case 2
        N=4*L-2;H=zeros(1,N);H(1:L)=1;H(end-L+2:end)=1;
    case 3
        N=4*L;H=zeros(1,N);H(1:L+1)=1;H(end-L+2:end)=1;
    otherwise
        disp('no such type');
end
h=ifft(H,N);
switch id
    case 1
        win=conv(boxcar(N),boxcar(N));wn='n';
    case 2
        win=conv(triang(N),boxcar(N));wn='s';
    case 3
        win=conv(triang(N),triang(N));wn='d';
    otherwise
```

```
        error('bad parameters');
end
win =win/max(win);hn = [h(2:end),h].* win';h =ifft(H,N);
zeroh =roots(fliplr(h));zerohn =roots(fliplr(hn));
zerowin =roots(fliplr(win));znh =zerosNum(zeroh);
znhn =zerosNum(zerohn);znwin =zerosNum(zerowin);
theta =0:0.01* pi:2* pi;mu =ones(size(theta));
if tp >1 ||id >1
    figure
end
subplot(1,3,1),hp(1) =plot(zeroh,'ko');hold on;
hp(2) =polar(theta,mu,'k--');
mkTxt(gca,zeroh,znh,strcat(tn,strcat(apt{1},'with \itN \rm =',num2str(N))));
subplot(1,3,2),hp(3) =plot(zerowin,'ko');
hold on,hp(4) =polar(theta,mu,'k--');
mkTxt(gca,zerowin,znwin,strcat(tn,apt{2},wn,' \rmwith \itN \rm =',num2str(N)));
subplot(1,3,3),hp(5) =plot(zerohn,'ko');
hold on,hp(6) =polar(theta,mu,'k--');
mkTxt(gca,zerohn,znhn,strcat(tn,apt{3},wn,'with \itN \rm =',num2str(N)));
set(hp,'MarkerSize',10,'MarkerEdgeColor','k','LineWidth',3);

functionznh =zerosNum(inp)
len =length(inp);znh =ones(1,len);
for m =1:len -1
    for n =m +1:len
if abs(real(inp(m) -inp(n))) <10^(-4)&&abs(imag(inp(m) -inp(n))) <10^(-4)
            znh(m) =znh(m) +1;
            inp(n) =inf;
        end
    end
end
znh(inp = =inf) =0;

functionmkTxt(ah,zt,zn,tln)
set(ah,'FontSize',16);grid(ah,'on');axis equal;
L =(zn ~ =0);LN =length(L);
for n =1:LN
    if L(n) ~ =0
      text(real(zt(n)) +0.1,imag(zt(n)),{num2str(zn(n))},'FontSize',16);
    end
end
```

```
text(-0.1,0,num2str(sum(zn)),'FontSize',16);
set(ah,'XLim',[-1.5 1.5],'YLim',[-1.5,1.5]);
xlabel(ah,'real part of \it \bfz_k');
ylabel(ah,'imag part of \it \bfz_k');
title(ah,tln);
```

由上面的分析易知，长度为 $2N-1$ 的 ap 半带滤波器具有 $2N-2$ 个零点，且均以倒数对的形式存在，一半分布在单位圆内，另一半分布在单位圆外。这样，单位圆内 $N-1$ 个零点可生成长度为 N 的滤波器 $H_0(z)$，单位圆外的 $N-1$ 个零点生成的滤波器则可记为 $H_0(z^{-1})$，ap 半带滤波器 $H(z)$ 可写成如下形式：

$$H(z) = \sigma^2 \prod_{k=1}^{2N-2}(1 - z_k z^{-1}) = \sigma^2 \prod_{k=1}^{N-1}(z_k - z^{-1}) \prod_{k=1}^{N-1}(z_k^{-1} - z^{-1}) = H_0(z)H_0(z^{-1}) \tag{7-59}$$

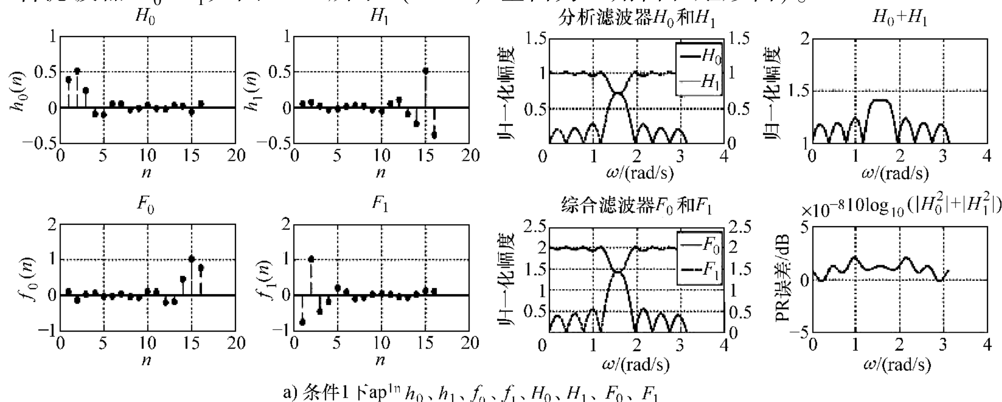
式（7-59）中，

$$H_0(z) = \sigma_x \prod_{k=1}^{N-1}(z_k - z^{-1}),\ H_0(z^{-1}) = \sigma_x \prod_{k=1}^{N-1}(z_k - z) \tag{7-60}$$

令 $z=1$，结合式（7-59）和式（7-60）得：

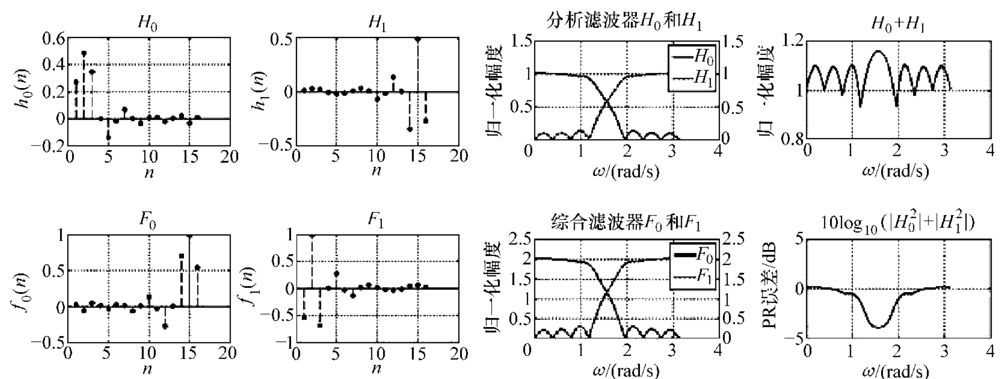
$$\sigma^2 \prod_{k=1}^{N-1}(-z_k^{-1})(1-z_k)^2 = \sigma_x^2 \prod_{k=1}^{N-1}(z_k-1)^2 \Rightarrow \sigma_x = \sigma\sqrt{\prod_{k=1}^{N-1}(-z_k^{-1})} \tag{7-61}$$

即把半带滤波器 $\boldsymbol{H}$ 进行分解后找到所有圆内的零点就可以确定分析低通滤波器 $\boldsymbol{h}_0$，再按照 PR 条件即式（7-34）关系依次求出分析高通滤波器 $\boldsymbol{h}_1$，综合滤波器 $\boldsymbol{f}_1$ 和 $\boldsymbol{f}_2$。但对于 $\mathrm{ap}^{1\mathrm{s}}$ 分布在单位圆上的零点，除了 ±1 点外都是单点，如果仍然按照式（7-59）分解则得到的 $\boldsymbol{h}_0$ 不是实数。由图（7-22）观察 3 种条件下 $\mathrm{ap}^{1\mathrm{s}}$ 的零点，单位圆上零点位置相近，因此，这里采纳隔 1 取 1 的方式形成 $\boldsymbol{H}_0$，这样 $\boldsymbol{\sigma}$ 的形成也要进行调整。在满足条件 1、2、3 下，$\mathrm{ap}^{1\mathrm{nsd}}$ 分析滤波器 $\boldsymbol{H}_0/\boldsymbol{H}_1$ 和综合滤波器 $\boldsymbol{F}_0/\boldsymbol{F}_1$ 如图 7-23 所示（$L=4$，基窗为三角窗和矩形窗）。

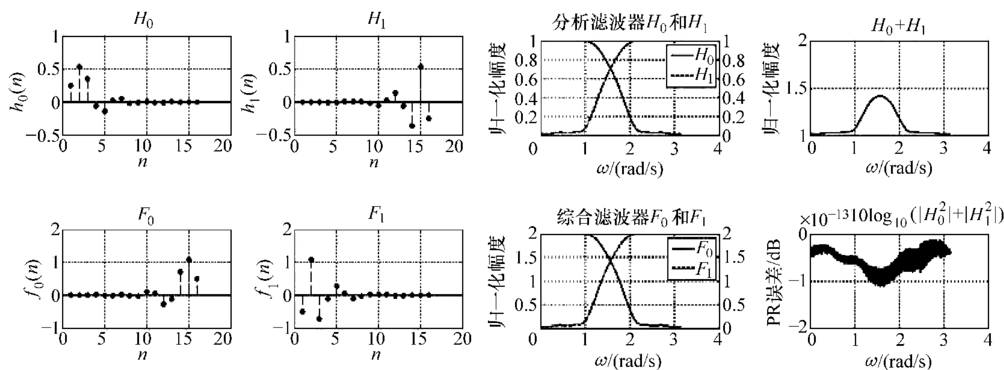


a) 条件1下$\mathrm{ap}^{1\mathrm{n}}$ h_0、h_1、f_0、f_1、H_0、H_1、F_0、F_1

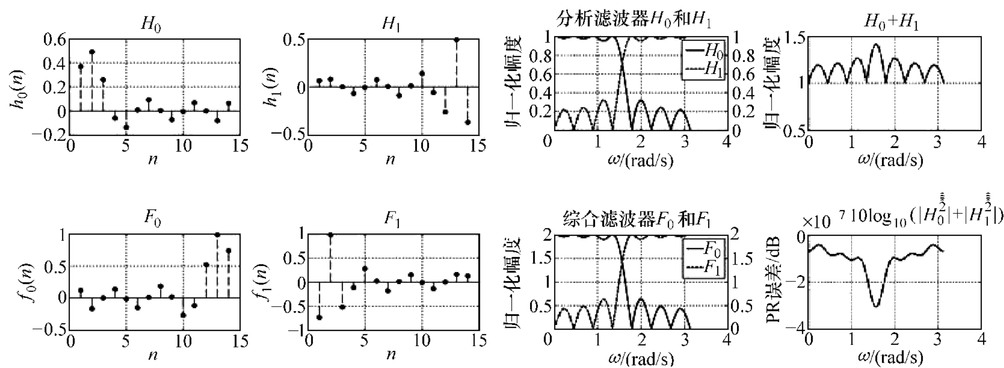
图 7-23　条件 1、2、3 下 $\mathrm{ap}^{1\mathrm{nsd}}$ 分析和综合滤波器



b) 条件1下ap[1s] h_0、h_1、f_0、f_1、H_0、H_1、F_0、F_1

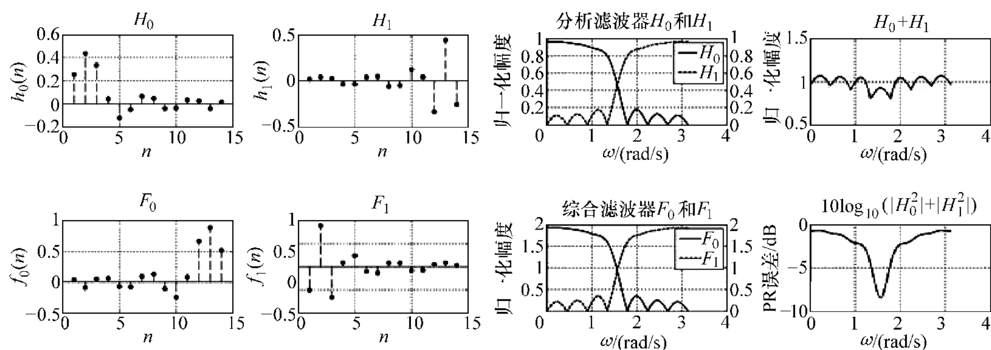


c) 条件1下ap[1d] h_0、h_1、f_0、f_1、H_0、H_1、F_0、F_1

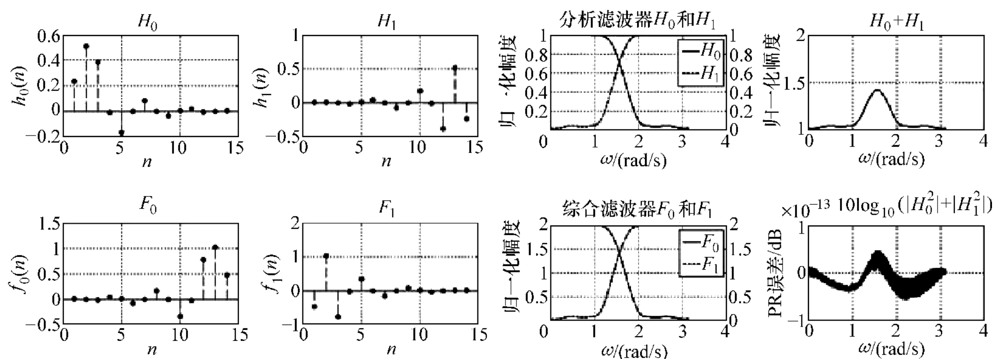


d) 条件2下ap[1n] h_0、h_1、f_0、f_1、H_0、H_1、F_0、F_1

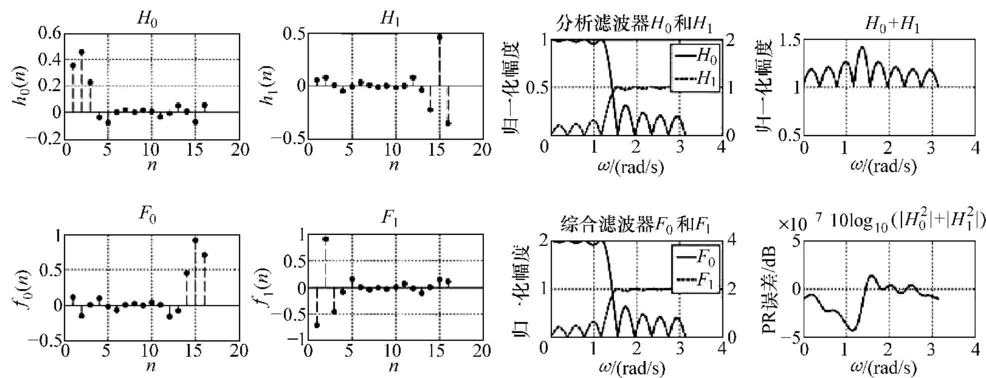
图 7-23　条件 1、2、3 下 ap[1nsd]分析和综合滤波器（续）



e) 条件2下ap^{1s} h_0、h_1、f_0、f_1、H_0、H_1、F_0、F_1

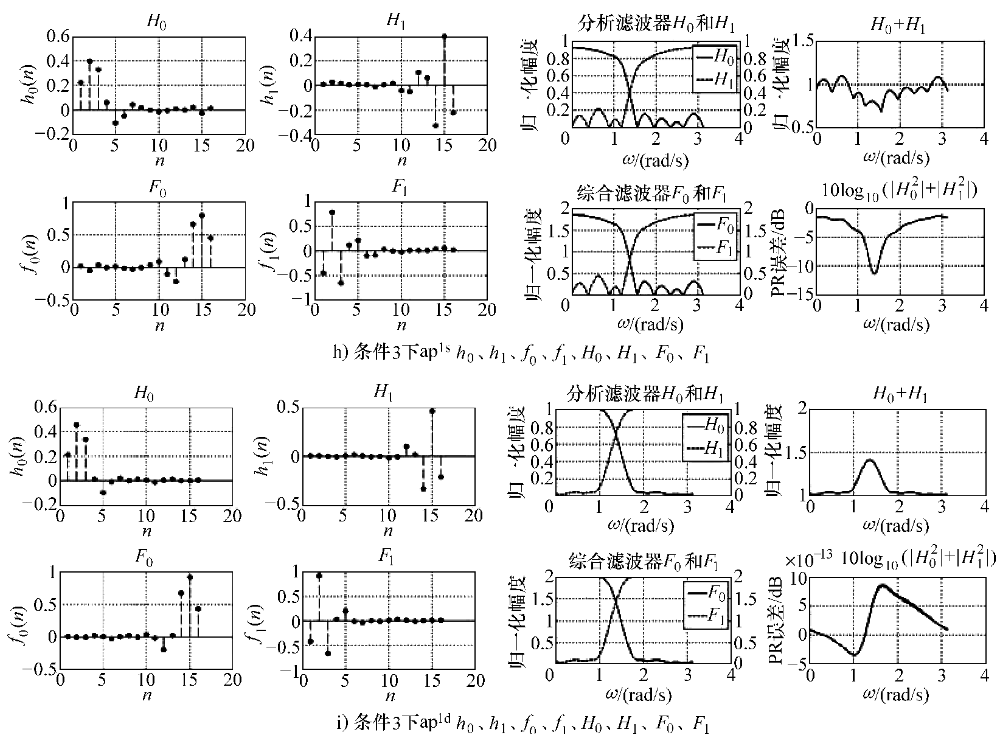


f) 条件2下ap^{1d} h_0、h_1、f_0、f_1、H_0、H_1、F_0、F_1



g) 条件3下ap^{1n} h_0、h_1、f_0、f_1、H_0、H_1、F_0、F_1

图 7-23　条件 1、2、3 下 ap^{1nsd}分析和综合滤波器（续）



h) 条件3下ap[1s] h_0、h_1、f_0、f_1、H_0、H_1、F_0、F_1

i) 条件3下ap[1d] h_0、h_1、f_0、f_1、H_0、H_1、F_0、F_1

图 7-23　条件 1、2、3 下 ap[1nsd]分析和综合滤波器（续）

分析图 7-23 中的实验结果，我们可以得到如下结论：3 种条件下，ap[1n]的重构误差精度达到 $10^{-8}\sim10^{-7}$dB，ap[1d]的重构误差则达到了 10^{-13}dB，ap[1n]的重构误差精度最低，但从子带互补性方面即 $\boldsymbol{H}_0+\boldsymbol{H}_1$，单窗具有最低泄露；横向对比，条件 1 下滤波器特性优于条件 2，而条件 2 优于条件 3。实现上述实验的 MATLAB 代码如下：

```
function spDiv(L)

if nargin = =0
    L =4;TN ={ 'Cond 1','Cond 2','Cond 3'};
    apt ={ '\it \bfH \rm_0','\it \bfC_','ap^1^'};
end
clc,close all
for tp =3:3
    for id =1:3
        spDIV(L,tp,id,TN{ tp },apt);
    end
end
```

```
function spDIV(L,tp,id,tn,apt)
switch tp
    case 1
        N =4* L;H =zeros(1,N);H(1:L) =1;H(L +1) =0.5;
        H(end -L +2:end) =1;H(end -L +1) =0.5;
    case 2
        N =4* L -2;H =zeros(1,N);H(1:L) =1;H(end -L +2:end) =1;
    case 3
        N =4* L;H =zeros(1,N);H(1:L +1) =1;H(end -L +2:end) =1;
    otherwise
        disp('no such type');
end
h =ifft(H,N);
switch id
    case 1
        win =conv(boxcar(N),boxcar(N));
    case 2
        win =conv(triang(N),boxcar(N));
    case 3
        win =conv(triang(N),triang(N));
    otherwise
        error('bad parameters');
end
win =win/max(win);hn =[h(2:end),h].* win';
h0 =mkH0(hn,tp,id);[h1,f0,f1] =mkh1f01(h0);
toDraw([h0;h1;f0;f1],tp,id);

function toDraw(h,tp,id)
y1n ={ '\it \bfh \rm_0(\itn \rm)','\it \bfh \rm_1(\itn \rm)','\it \bff \rm_0(\itn \rm)',
'\it \bff \rm_1(\itn \rm)'};
t1n ={ '\it \bfH \rm_0','\it \bfH \rm_1','\it \bfF \rm_0','\it \bfF \rm_1'};
N =size(h,2);w =0:2* pi/N^4:pi;
expnw =exp( -j* w'* [0:N -1]);

if tp +id > 2
    figure
end
for i =1:4
    subplot(2,2,i),stem(real(h(i,:)),'Fill','k --','MarkerSize',6);
    set(gca,'FontSize',16);xlabel('\itn');ylabel(y1n{ i });
    title(t1n{ i });grid on;
```

```
    hns = repmat(h(i,:),length(w),1);
    HW(:,i) = sum(expnw.* hns,2);
end

figure,HW = abs(HW);
subplot(2,2,1),[ax,h1,h2] = plotyy(w,HW(:,1),w,HW(:,2));
set(h1,'LineStyle','-','Color','k','LineWidth',2);
set(h2,'LineStyle','--','Color','k','LineWidth',2);
set(ax ,'FontSize',16,'YColor','k');
grid on;legend('\it \bfH \rm_0','\it \bfH \rm_1');
xlabel('\it \omega(rad/s)');ylabel('amp');
title('Analysis Filter \it \bfH \rm_0 and \it \bfH \rm_1');
subplot(2,2,2),plot(w,HW(:,1) + HW(:,2),'k','LineWidth',2);
set(gca ,'FontSize',16);grid on;xlabel('\it \omega(rad/s)');title('\it \bfH \rm_
0 + \it \bfH \rm_1');ylabel('amp');
subplot(2,2,3),[ax,h1,h2] = plotyy(w,HW(:,3),w,HW(:,4));
set(h1,'LineStyle','-','Color','k','LineWidth',2);
set(h2,'LineStyle','--','Color','k','LineWidth',2);
set(ax ,'FontSize',16,'YColor','k');
grid on;legend('\it \bfF \rm_0','\it \bfF \rm_1');
xlabel('\it \omega(rad/s)');ylabel('amp');
title('Synthesis Filter \it \bfF \rm_0 and \it \bfF \rm_1');
subplot(2,2,4),plot(w,10* log(HW(:,1).^2 + HW(:,2).^2),'k','LineWidth',2);set
(gca ,'FontSize',16);grid on;
xlabel('\it \omega(rad/s)');ylabel('PR error(dB)');
title('10log_1_0(|\it \bfH \rm_0^2 |+ |\it \bfH \rm_1^2 |)');

function h0 = mkH0(h,tp,id)
zeroh0 = roots(fliplr(h));len = length(zeroh0);zeroh00 = zeroh0;
h0 = zeros(1,len/2);k = 1;eo = 0;
for m = 1:len
    A = (zeroh0(m) ~= nan);B = (abs(abs(zeroh0(m)) -1) <= 10^(-6));
    C = (abs(zeroh0(m)) < 1);
    if A
        if(B ||((~B)&& C))
            h0(k) = zeroh0(m);
            zeroh0(abs(zeroh0 - h0(k)) < 10^(-6)) = nan;
            k = k +1;
        end
    end
end
```

```
if id==2
    sil=abs(abs(h0)-1)<10^(-6);zon=h0(sil);zin=h0(~sil);
    len=length(zon);m=1;
    for n=1:len
        if((mod(n,4)==1)||(mod(n,4)==2))
            zonout(m)=zon(n);m=m+1;
        end
    end
end
mu=sum(h)/prod(1-zeroh00);mux=sqrt(mu)* sqrt(prod(-1./h0));
if id==2
    h0=[zonout zin];mux=mux* 1.13;
end
if tp==3
    h0=mux* poly(h0);
else
    h0=real(mux* poly(h0));
end
function[h1,f0,f1]=mkh1f01(h0)
N=length(h0);
for k=1:N
    h1(k)=(-1)^(k+1)* h0(N+1-k)';
    f0(k)=2* h0(N+1-k)';
    f1(k)=2 * (-1)^(k)* h0(k);
end
```

7.3　图像子带分解

基于分解子带的图像编码称为频带分割编码，其基本思想是把信号分解成低分辨率的平滑近似和高频细节两部分，分别称为低频子带和高频子带。对每个生成的子带又可以用同样的方式进一步分解。对于图像的分解通过在水平和垂直方向上分解来实现。然后，对于不同分辨率的子带图像采用不同的量化精度和编码方法来达到图像压缩的目的。Mallat 在多分辨率分析理论的基础上，用子带结构实现离散小波变换的算法，统一了子带滤波与小波变换的计算，使小波变换成为图像处理的重要手段，为图像压缩和编码提供了有效的方法。

一幅图像可看成二维信号，经小波变换后产生了 4 个子带图像，即水平和垂直方向低频子带 LL，垂直方向低频和水平方向高频子带 HL，水平方向低频和垂直方向高频 LH 以及水平和垂直方向都是高频子带 HH。分解方式如图 7-24 所示。

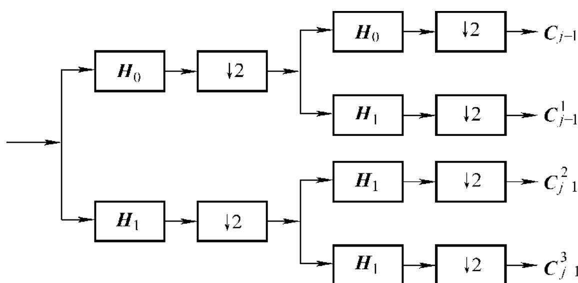


图 7-24　二维信号分解

图 7-24 中子带 $\boldsymbol{C}_{j-1}$集中了图像的大部分能量，大部分幅值较大的系数集中在该子带中，以后的小波变换都是对前次变换产生的低频子带进行变换。图 7-25 说明了一幅图像经过二级小波变换的过程以及形成的各个子带分布。从图中可以看出，二维数字信号的离散小波变换最终转化为数字滤波。对于紧支小波，相应的滤波器类型是 FIR，滤波器系数由尺度函数与小波函数决定。某些情况下，滤波器的线性相位特性可避免不必要的失真。为了得到线性相位滤波器须要求 FIR 系数对称，为此需设计对称的小波函数。另一方面，小波基的正交性限制了所能得到的小波基类型，特别是除了 Harr 小波外，没有任何紧支正交小波具有对称性。为此，考虑设计基于 ap^{2}的图像分解和重构。

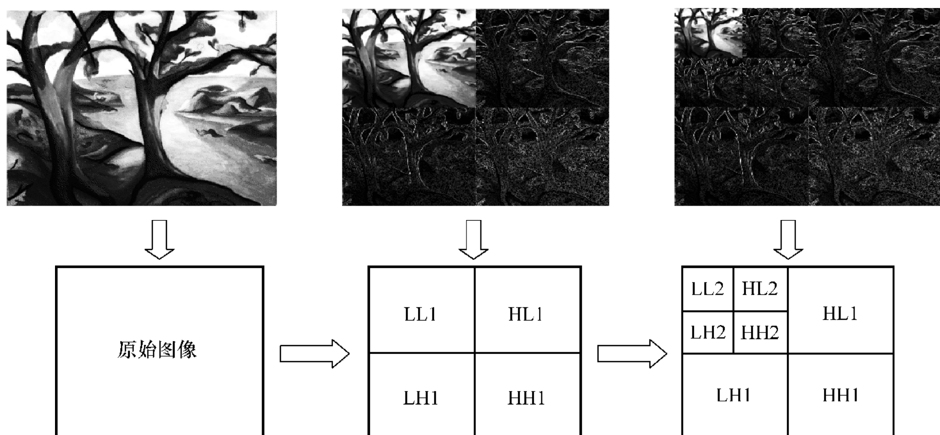


图 7-25　图像的子带分解（db 小波）

图像的子带分解及重构对应二维四子带 QMFB，结合图 7-9 和图 7-18，系统原理图如图 7-26 所示。

由图 7-26 可知：

$$V_i(z_r, z_c) = H_i(z_r, z_c)X(z_r, z_c)\quad i = 0, 1, 2, 3 \tag{7-62}$$

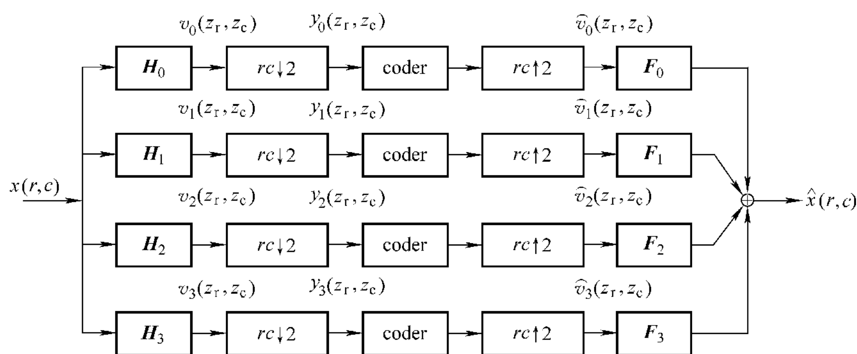


图 7-26　二维四子带 QMFB

经二维 2：1 下取样得：

$$Y_i(z_r,\ z_c) = \frac{1}{4}\sum_{k_r=0}^{1}\sum_{k_c=0}^{1} H_i(z_r^{1/2}W^{k_r},\ z_c^{1/2}W^{k_c})X_i(z_r^{1/2}W^{k_r},\ z_c^{1/2}W^{k_c})\ i = 0,1,2,3 \tag{7-63}$$

经综合滤波器得：

$$\widehat{X}(z_r,\ z_c) = \sum_{i=0}^{3} F_i(z_r,\ z_c)\widehat{V}_i(z_r,\ z_c) = \sum_{i=0}^{3} F_i(z_r,\ z_c)\hat{Y}_i(z_r^2,\ z_c^2) \tag{7-64}$$

若忽略编码误差，将式（7-63）代入到式（7-64）中并整理得：

$$\hat{X}(z_r,\ z_c) = \frac{1}{4}\sum_{i=0}^{3}\sum_{k_r=0}^{1}\sum_{k_c=0}^{1} F_i(z_r,\ z_c)H_i(z_rW^{k_r},\ z_cW^{k_c})X(z_rW^{k_r},\ z_cW^{k_c}) \tag{7-65}$$

定义如下矩阵和向量：

$$\boldsymbol{H}(z_r,z_c) = \begin{bmatrix} H_0(z_r,z_c) & H_0(z_r,\ -z_c) & H_0(-z_r,z_c) & H_0(-z_r,\ -z_c) \\ H_1(z_r,z_c) & H_1(z_r,\ -z_c) & H_1(-z_r,z_c) & H_1(-z_r,\ -z_c) \\ H_2(z_r,z_c) & H_2(z_r,\ -z_c) & H_2(-z_r,z_c) & H_2(-z_r,\ -z_c) \\ H_3(z_r,z_c) & H_3(z_r,\ -z_c) & H_3(-z_r,z_c) & H_3(-z_r,\ -z_c) \end{bmatrix} \tag{7-66}$$

$$\boldsymbol{F}(z_r,z_c) = [F_0(z_r,z_c)\quad F_1(z_r,z_c)\quad F_2(z_r,z_c)\quad F_3(z_r,z_c)]$$

$$\boldsymbol{X}(z_r,z_c) = [X(z_r,z_c)\quad X(z_r,\ -z_c)\quad X(-z_r,z_c)\quad X(-z_r,\ -z_c)]^{\mathrm{T}}$$

$$\boldsymbol{T}(z_r,z_c) = c\,[z_r^{-n_r}z_c^{-n_c}\quad 0\quad 0\quad 0]^{\mathrm{T}}$$

其中 H 称为重叠矩阵，式（7-65）可重写成如下：

$$\hat{X}(z_r,\ z_c) = \frac{1}{4}\boldsymbol{F}(z_r,\ z_c)\boldsymbol{H}(z_r,\ z_c)\boldsymbol{X}(z_r,\ z_c) \tag{7-67}$$

若完全消除混叠失真，要求：

$$\boldsymbol{F}(z_r,\ z_c)\boldsymbol{H}(z_r,\ z_c) = \boldsymbol{T}(z_r,\ z_c) \tag{7-68}$$

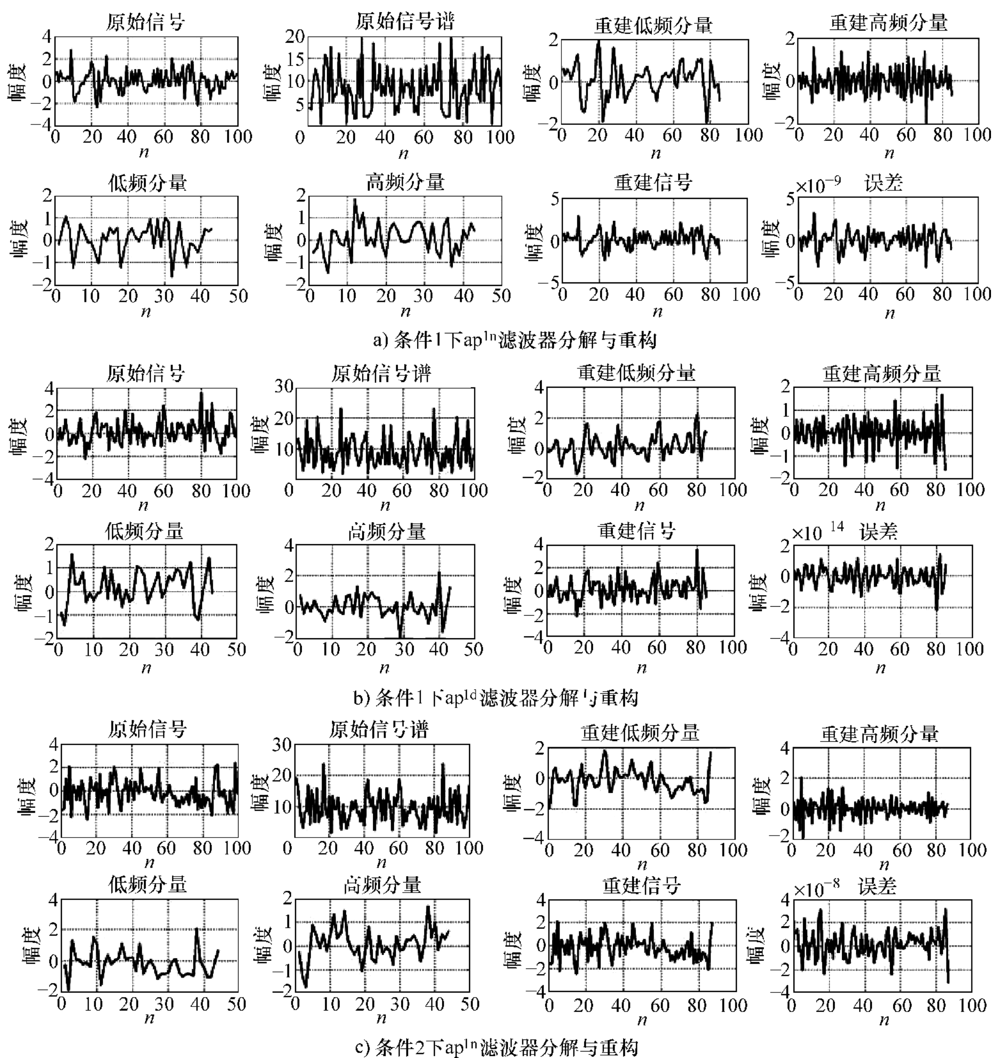
这样，就可以由分析滤波器求出综合滤波器 $\boldsymbol{F}$，但结果难以保证是 FIR 滤波器

组，利用多相分解可以解决，但过程复杂。实际上，对于二维图像的分解可通过一维滤波器分别在行列方向上进行。

7.4　全相位 2QMF 多分辨率实验

7.4.1　一维信号分解与重构

对高斯白噪声分别采用图 7-23 中 3 种条件下 ap^{1nsd} 滤波器进行分解和重构，取样点数为 100，分解得到的低频、高频及重构信号和误差结果如图 7-27 所示。

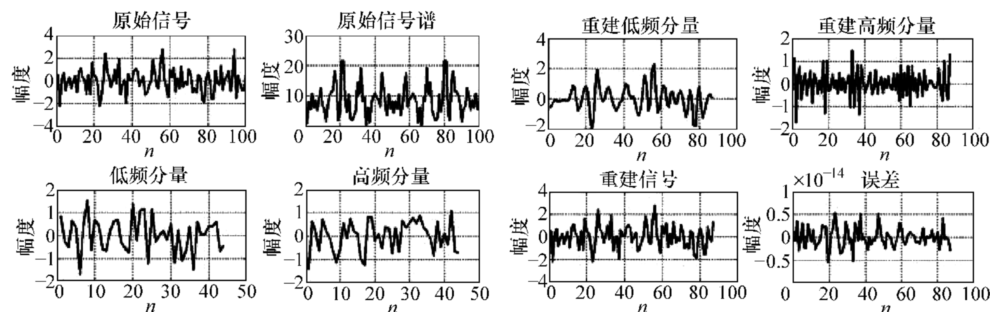


a) 条件1下ap^{1n}滤波器分解与重构

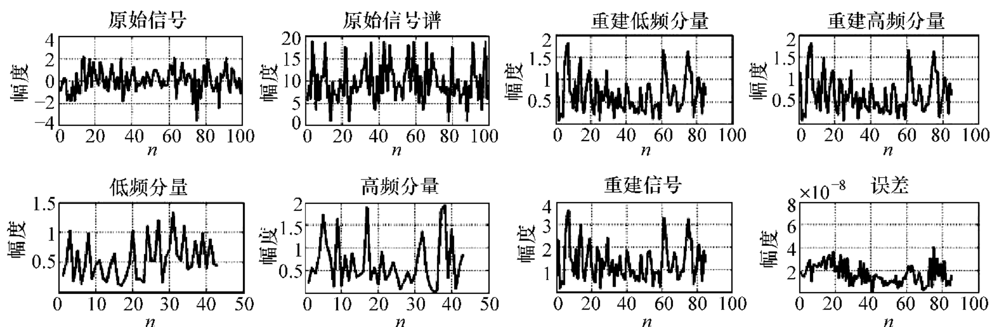
b) 条件1下ap^{1d}滤波器分解与重构

c) 条件2下ap^{1n}滤波器分解与重构

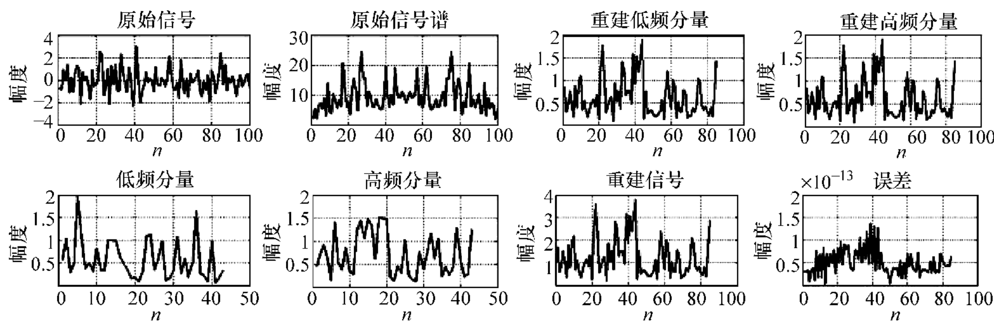
图 7-27　ap^{1nd}设计的 2QMFB 分析和重构试验



d) 条件2下ap^{1d}滤波器分解与重构



e) 条件3下ap^{1n}滤波器分解与重构



f) 条件3下ap^{1d}滤波器分解与重构

图 7-27　ap^{1nd}设计的 2QMFB 分析和重构试验（续）

上面的实验结果验证了 ap^{1nd}设计 2QMFB 的正确性。需要说明的是，第三类条件设计的滤波器组不具有线性相位，因此只对幅度误差进行了分析。结果显示，ap^{1n}重构的误差为 10^{-8}，而 ap^{1d}则达到了 10^{-14}左右，基本达到完全重构的目标。3 种条件下的 ap^{1s}，重构误差则较大，ap^{1s}条件 1 下的实验结果如图 7-28 所示。

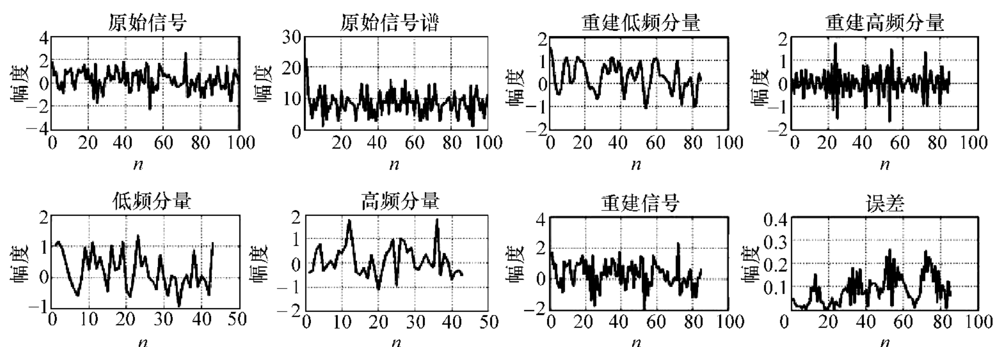


图 7-28　ap^{1s}设计的 2QMFB 分析和重构试验

实现以上实验结果的 MATLAB 代码如下（省去绘图部分）：

```
function res = s1AS(L)
if nargin = =0
    L = 4;NN = 100;
end
clc,close all;
asf = spDiv(L);sig = randn(1,NN);

for tp = 1:1
    for id = 2:2
        [L H C E err] = toProcess(asf{ tp }{ id },sig,tp);
        toDraw(L,H,C,E,err,sig);
    end
end

function[Lres Hres Cres Eres err] = toProcess(flt,in,tp)
h0 = fliplr(flt{ 1 });h1 = fliplr(flt{ 2 });
f0 = fliplr(flt{ 3 });f1 = fliplr(flt{ 4 });

N = length(flt{ 1 });sig = [fliplr(in(1:N -1)),in];
len_sig = length(sig);
for n = 1:len_sig - N +1
    L(n) = sum(h0. * sig(n:n +N -1));
    H(n) = sum(h1. * sig(n:n +N -1));
end
err = L + H;Lres = L(N:2:end);Hres = H(N:2:end);
L(2:2:end) = 0;H(2:2:end) = 0;

L = [fliplr(L(1:N -1)),L];H = [fliplr(H(1:N -1)),H];
```

```
for n =1:len_sig -N +1
    C(n) =sum(f0.* L(n:n +N -1));
    E(n) =sum(f1.* H(n:n +N -1));
end

Cres =C(N:end);Eres =E(N:end);
err =abs((Cres +Eres) -in(1:end -N +1));
if tp = =3
    Lres =abs(Lres);Hres =abs(Hres);Cres =abs(Cres);Eres =abs(Cres);
end
```

ap^1设计得到的 2QMFB 最大的优点是重建误差精度高，但也存在 3 个问题：一是恢复得到的信号有 $N-1$ 个延迟，从上面的程序可以看出；其二是运算过程涉及零点统计与谱分解，比较复杂；其三是形成的高低通滤波器 $\boldsymbol{H}_0$ 和 $\boldsymbol{H}_1$ 并不满足子带互补，这由图 7-23 实验结果得到证实。实际上，子带互补和功率互补不可能同时实现，矛盾之处在于过渡带。实际上，按照 $ap^{1\,nsd}$ 很容易形成严格互补的半带滤波器组 $\boldsymbol{H}_0$，$\boldsymbol{H}_1$。若信号频谱分布在 $\boldsymbol{\pi}/2$ 处分量较少，则可采用 $\boldsymbol{H}_0$ 和 $\boldsymbol{H}_1$ 进行分解和重构。如采用 $N=64$ 设计得到的 $ap^{1\,d}$ 高低通半带滤波器及特性如图 7-29 所示。

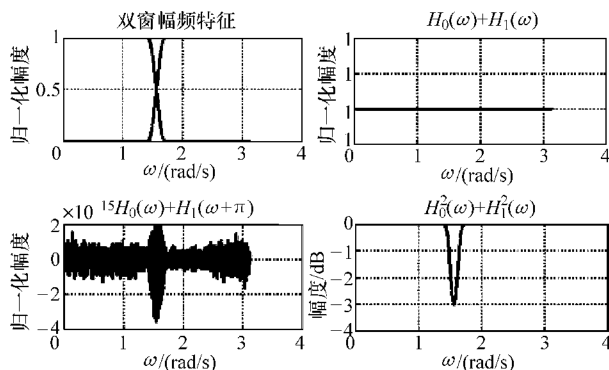


图 7-29　条件 1 下 $ap^{1\,d}$ 半带高低通滤波器

由图 7-29 可以看到，$ap^{1\,d}$ 高低通半带滤波器是严格子带互补的，且成镜像对称关系，而功率谱则在过渡带出现 3dB 误差。利用此滤波器组针对长度为 200、占空比为 50% 的方波进行分析和重构的结果如图 7-30 所示。

由图中可见，重构误差小于 2%，通过增加滤波器阶数可进一步降低重构误差。此外，还可以根据信号谱分布的规律，预先过滤出 $\boldsymbol{\pi}/2$ 分量。由图 7-29 可知剩下的信号分量可无失真重构，而对于 $\boldsymbol{\pi}/2$ 分量则按照带宽的占比进行高比下取样。对比谱分解，此方法的优点有二：一是构造简单，按照条件 1/2/3 的抽

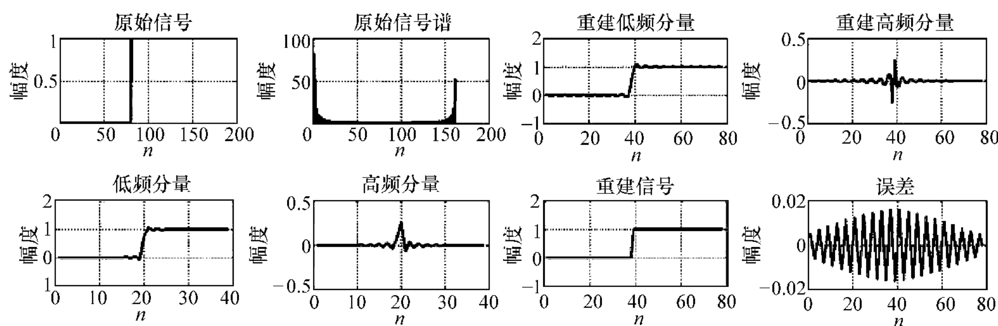


图 7-30　ap[1d]半带高低通滤波器分析与重构

样序列 $\boldsymbol{H}$ 进行设计即可得到分析和综合滤波器，无需谱分解；二是高低通滤波器严格子带互补，因此两通道完全包含了信号全部谱信息，这点很重要。尤其对于不是以重构为目标的应用如信号分析，子带提取等方面相对 ap QMFB 和小波来说具有一定优势。实现图 7-29 和图 7-30 实验结果的 MATLAB 代码（绘图部分省略）如下：

```
function qmfbA(N)
if nargin==0
    N=64;n=1/4;
end
clc,clf;
H=zeros(1,N);H(1:N*n)=1;H(N*n+1)=0.5;
H(N-N*n+2:N)=1;H(N-N*n+1)=0.5;
apFNSD(N,H,'d');

function out=apFNSD(N,H,wt)

HL=H;HH=ones(size(H))-HL;
hL=ifft(HL);hH=ifft(HH);
% construct COS
w=0:pi/(N*N):pi;cosw=w'*[0:N-1];
wpi=pi:pi/(N*N):2*pi;coswpi=wpi'*[0:N-1];
% constuct convulated window
switch wt
    case{'n','N'}
        wb=boxcar(N);wf=wb;wn='none window';
    case{'s','S'}
        wb=boxcar(N);wf=triang(N);wn='single window';
    case{'d','D'}
        wb=triang(N);wf=wb;wn='dual window';
```

```
    otherwise
        disp('NO SUCH WINDOW');
end
% normalized convulated window
wconv=conv(wb,wf);C=wconv/wconv(N);cn=C(N:2*N-1);
hgL=sum(2*cos(cosw)*diag(cn.*hL'),2);hgL=hgL-C(N)*hL(1);
hgH=sum(2*cos(cosw)*diag(cn.*hH'),2);hgH=hgH-C(N)*hH(1);
hgHL=sum(2*cos(coswpi)*diag(cn.*hH'),2);hgHL=hgHL-C(N)*hH(1);

function res=s1AS2(L)

if nargin==0
    LL=16;NN=200;
end
clc,close all;n=[0:NN-1];
asf=spDiv(LL);sig=zeros(1,NN);sig(NN/2:NN)=1;
for tp=2:2
    for id=3:3
        [LH]=mkFlt(tp,id,LL);[L,H,C,E,err]=toProcess(L,H,sig);
        N=(length(L)+1)/2;
        toDraw(L,H,C,E,err,sig(N:end-N+1));
    end
end

function[LL HH,MM]=mkFlt(tp,id,L)
alpha=0.5;
switch tp
    case 1
        N=4*L;L=L+1;H=zeros(1,N);H(1:L)=1;H(L+1)=alpha;
        H(end-L+2:end)=1;H(end-L+1)=alpha;
    case 2
        N=4*L-2;H=zeros(1,N);H(1:L)=1;H(end-L+2:end)=1;
    case 3
        N=4*L;H=zeros(1,N);H(1:L+1)=1;H(end-L+2:end)=1;
    otherwise
        disp('no such type');
end
h0=ifft(H,N);HH=1-H;h1=ifft(HH,N);
switch id
    case 1
        win=conv(boxcar(N),boxcar(N));
```

```
    case 2
        win = conv(triang(N),boxcar(N));
    case 3
        win = conv(triang(N),triang(N));
    otherwise
        error('bad parameters');
end
win = win/max(win);LL = [h0(2:end),h0].* win';
HH = [h1(2:end),h1].* win';

function[Lres Hres Cres Eres err] = toProcess(h0,h1,in)
N = (length(h0) +1)/2;
sig = [fliplr(in(1:N -1)),in,fliplr(in(end -N +2:end))];
len_sig = length(sig);
for n = N:len_sig -N +1
    L(n -N +1) = sum(h0.* sig(n -N +1:n +N -1));
    H(n -N +1) = sum(h1.* sig(n -N +1:n +N -1));
end
Lres = L(N:2:end -N +1);Hres = H(N:2:end -N +1);
L(2:2:end) =0;H(2:2:end) =0;

L = [fliplr(L(1:N -1)),L,fliplr(L(end -N +2:end))];
H = [fliplr(H(1:N -1)),H,fliplr(H(end -N +2:end))];
for n = N:len_sig -N +1
    C(n -N +1) =2* sum(h0.* L(n -N +1:n +N -1));
    E(n -N +1) =2* sum(h1.* H(n -N +1:n +N -1));
end

Cres = C(N:end -N +1);Eres = E(N:end -N +1);
err =  Cres +Eres -in(N:end -N +1);
```

7.4.2　二维信号分解与重构

二维信号的分解和重构可以采用两种 ap 方式来进行，即基于 2QMFB 和基于 ap 半带滤波器。由 7.3 节分析结论可知，二维信号 2QMFB 处理可在行列方向上分别进行处理，即相同的高低通滤波器先在行方向分解然后在 2∶1 下采样，再逐列进行分解并 2∶1 下取样，从而得到四个子带；重构则按照相反顺序进行处理。图 7-31 是实验结果（$\boldsymbol{H}_0$按照条件 2 $\mathrm{ap}^{1\mathrm{d}}$设计得到，$N = 16$）。

在二维图像多分辨率分析中，ap 功率互补滤波器组表现得并不十分理想，这是因为行列分别进行了不相关处理，对于零值行和列内插时只能通过列向或行向单独进行，才可以利用此结果进行行或列方向上其他零点的内插，显然有误差

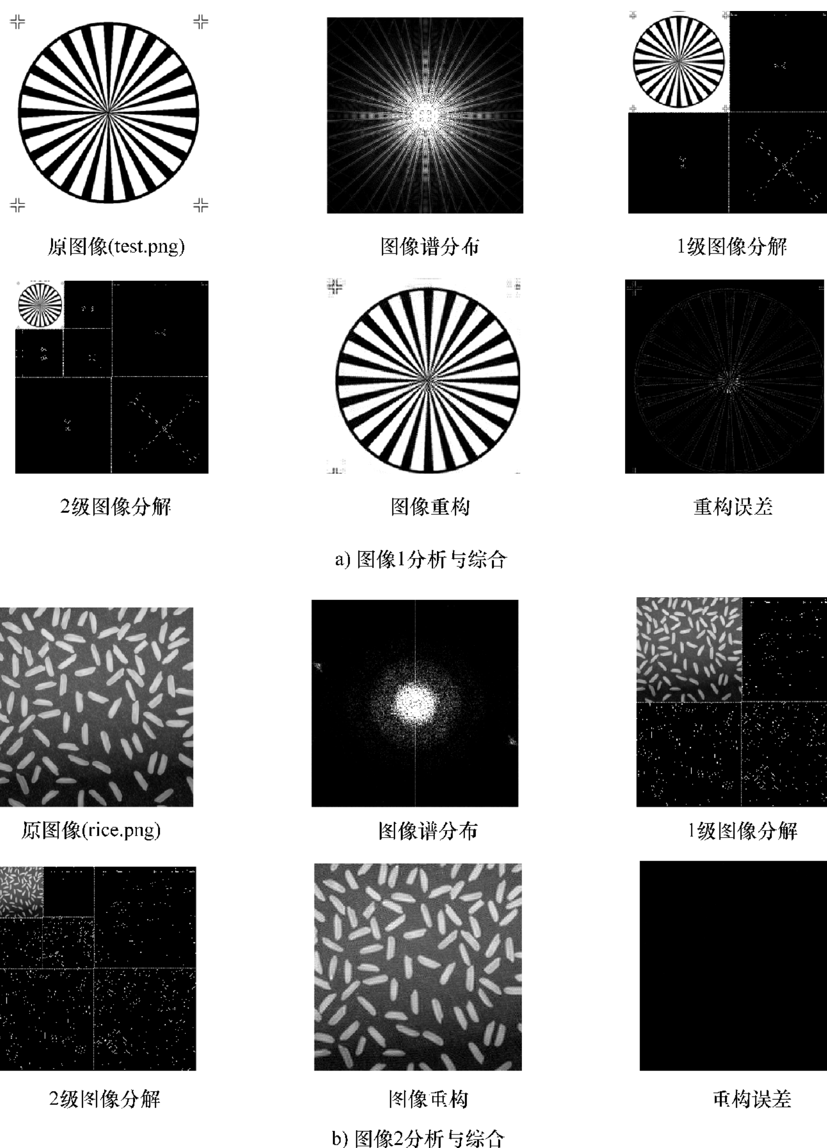


图 7-31　基于条件 2 下 ap^{1d} 2QMFB 的图像分析

累积。这时，ap 功率互补滤波器组的非子带互补性质也造成谱的非正交分割。观察图 7-31 针对图 1 的实验，原图像具有丰富的高低频分量。由一级和二级分解看出，LH 和 HL 子图内容并未充分体现原图分量，尽管主要能量集中在中低频，高频分析子图效果良好。图像 2 的能量主要集中在低频，列方向上谱比行方向上谱丰富，子图内容得以体现，重构效果明显优于图像 1。实现上述实验的

MATLAB 代码如下：

```
function res = s2AS1(L)

if nargin = =0
    L =4;
    clc,close all;
end
imgf ={ 'woman.bmp','test.png','kids.tif','rice.png','peppers.png',...
        'pears.png','fabric.png'};
path(path,'D:\work\book1\chapter 7\images');asf =spDiv(L);
[X,XFT,LL,HL,LH,HH,LL2,HL2,LH2,HH2,CC,EE] =toProcess2(asf{2}{3},imgf);
toDraw(X,XFT,LL,HL,LH,HH,LL2,HL2,LH2,HH2,CC,EE);

function[X,XFT,LL HL LH HH LL2 HL2 LH2 HH2  CC EE] =toProcess2(flt,in)
[X XFT h0 h1 f0 f1] =preProcess(flt,in);
[LL0,LH0,HL0,HH0,LL,LH,HL,HH] =apAnalysis(h0,h1,X);
[CC,EE] =apSynthesis(f0,f1,LL0,LH0,HL0,HH0,X);
[LL0,LH0,HL0,HH0,LL2,LH2,HL2,HH2] =apAnalysis(h0,h1,LL);

function[X XFT h0 h1 f0 f1] =preProcess(flt,in)
h0 =fliplr(flt{ 1 });h1 =fliplr(flt{ 2 });
f0 =fliplr(flt{ 3 });f1 =fliplr(flt{ 4 });k =4;
imgd =imread(in{ k });imginfo =imfinfo(in{ k });
ifndims(imgd) = =3
    X =uint8(0.30* imgd(:,:,1) +0.59* imgd(:,:,2) +0.11* imgd(:,:,3));
else
    X =imgd(:,:);
end
C =imginfo.Width;R =imginfo.Height;N =length(h0);

XFT =fftshift(abs(fft2(X,R,C)));
XFT =(XFT -min(min(XFT)))/(max(max(XFT)) -min(min(XFT)))* 256;

function[LL0,LH0,HL0,HH0,LL,LH,HL,HH] =apAnalysis(h0,h1,Xin)
N =length(h0);[R C] =size(Xin);
Xext =extImg(Xin,N,R,C);[R C] =size(Xext);
for r =1:R
    for c =N:C
        H0(r,c -N +1) =sum(h1.* double(Xext(r,c -N +1:c)));
        L0(r,c -N +1) =sum(h0.* double(Xext(r,c -N +1:c)));
    end
```

```
end

for c=1:C-N+1
    for r=N:R
        LL0(r-N+1,c)=sum(h0'.*(L0(r-N+1:r,c)));
        LH0(r-N+1,c)=sum(h1'.*(L0(r-N+1:r,c)));
        HL0(r-N+1,c)=sum(h0'.*(H0(r-N+1:r,c)));
        HH0(r-N+1,c)=sum(h1'.*(H0(r-N+1:r,c)));
    end
end

LL1=uint8(LL0);LH1=uint8(LH0);HL1=uint8(HL0);HH1=uint8(HH0);

LL=LL1(1:2:end,1:2:end);LH=LH1(1:2:end,1:2:end);
HL=HL1(1:2:end,1:2:end);HH=HH1(1:2:end,1:2:end);
LH=(LH-min(min(LH)))/(max(max(LH))-min(min(LH)))*256;
HL=(HL-min(min(HL)))/(max(max(HL))-min(min(HL)))*256;
HH=(HH-min(min(HH)))/(max(max(HH))-min(min(HH)))*256;

function[CC,EE]=apSynthesis(f0,f1,LL0,LH0,HL0,HH0,X)
LL0(2:2:end,:)=0;LL0(:,2:2:end)=0;
LH0(2:2:end,:)=0;LH0(:,2:2:end)=0;
HL0(2:2:end,:)=0;HL0(:,2:2:end)=0;
HH0(2:2:end,:)=0;HH0(:,2:2:end)=0;
[R,C]=size(LL0);N=length(f0);
LLe=extImg(LL0,N,R,C);LHe=extImg(LH0,N,R,C);
HLe=extImg(HL0,N,R,C);HHe=extImg(HH0,N,R,C);

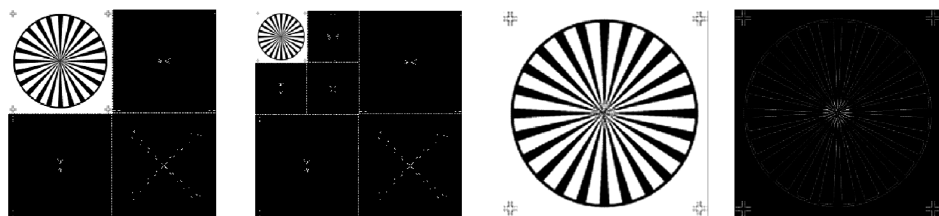
[R,C]=size(LLe);
for r=1:R
    for c=N:C
        LLe0(r,c-N+1)=(sum(f0.*double(LLe(r,c-N+1:c))));
        LHe0(r,c-N+1)=(sum(f1.*double(LHe(r,c-N+1:c))));
        HLe0(r,c-N+1)=(sum(f0.*double(HLe(r,c-N+1:c))));
        HHe0(r,c-N+1)=(sum(f1.*double(HHe(r,c-N+1:c))));
    end
end
for c=1:C-N+1
    for r=N:R
        LL2(r-N+1,c)=(sum(f0'.*(LLe0(r-N+1:r,c))));
        LH2(r-N+1,c)=(sum(f0'.*(LHe0(r-N+1:r,c))));
```

```
        HL2(r-N+1,c)=(sum(f1'.* (HLe0(r-N+1:r,c))));
        HH2(r-N+1,c)=(sum(f1'.* (HHe0(r-N+1:r,c))));
    end
end
CC=uint8(LL2(N:end,N:end)+LH2(N:end,N:end)+...
    HL2(N:end,N:end)+HH2(N:end,N:end));
EE=CC-X(1:end-N+1,1:end-N+1);

function imgout=extImg(imgin,N,r,c)
imgout=[fliplr(flipud(imgin(1:N-1,1:N-1))),imgin(1:N-1,:);fliplr(imgin
(:,1:N-1)),imgin];

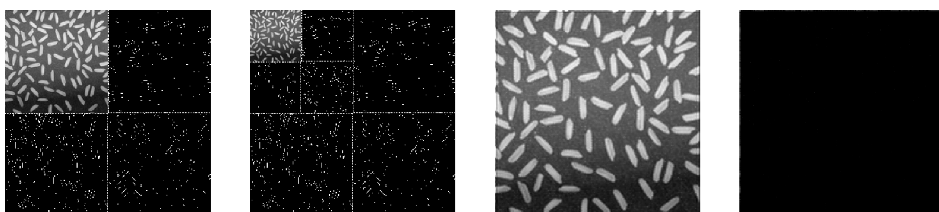
function toDraw(X,XFT,LL,HL,LH,HH,LL2,HL2,LH2,HH2,CC,EE)
imshow(X);figure(2),imshow(XFT);figure(3),
LH(1:2:end,1)=255;LH(end,1:2:end)=255;
HL(1,1:2:end)=255;HL(1:2:end,end)=255;
imshow([LL,LH;HL,HH]);figure(4),
LH2(1:2:end,1)=255;LH2(end,1:2:end)=255;
HL2(1,1:2:end)=255;HL2(1:2:end,end)=255;
imshow([[LL2,LH2;HL2,HH2],LH;HL,HH]);
figure(5),imshow(CC);figure(6),imshow(EE);
```

基于 ap 半带滤波器的实验结果如图 7-32 所示（条件与图 7-30 相同）。



1级图像分解　　2级图像分解　　重构图像　　重构误差

a) 图像1分析与综合



1级图像分解　　2级图像分解　　重构图像　　重构误差

b) 图像2分析与综合

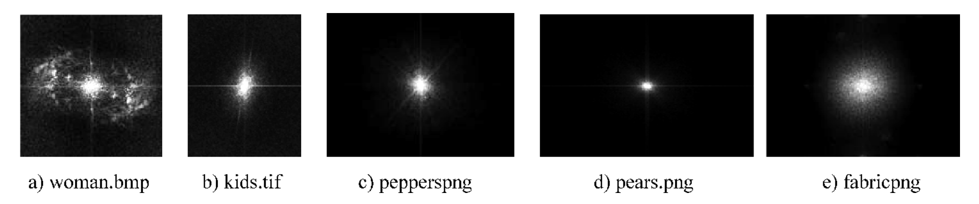
图 7-32　基于条件 2 下 ap^{1d} 半带滤波器的图像分析

图像 1 的行列谱在 π/2 频率处均有分量，因此重构效果明显劣于 ap 2QMFB；对于图像 2 的分析与重构则表现出与 ap 2QMFB 几乎相同的优良效果。关键是 ap 半带滤波器设计复杂度较低，且由第 2 章及第 4 章内容可知很方便地能设计出各种频率分辨率的谱分解，而且可以根据应用需求选择不同的正交基类型（MATLAB 代码基本与上 ap 2QMFB 相同）。表 7-1 列出了两种方法对不同图像重构的峰值信噪比。

表 7-1　ap 2QMFB 和 ap 半带滤波器重构图像 PSNR（dB）

方法＼图像	图像 1	图像 2	woman. bmp	kids. tif	peppers. png	pears. png	fabric. png
ap 2QMFB	32. 52	32. 35	29. 74	41. 76	40. 07	38. 84	33. 32
ap 半带	32. 10	32. 36	29. 02	39. 98	40. 03	38. 15	33. 59

各图像谱如图 7-33 所示。



a) woman.bmp　b) kids.tif　c) pepperspng　d) pears.png　e) fabricpng

图 7-33　受测图像谱分布

7. 5　小结

本章首先介绍了一维和二维信号抽取和内插信号谱变化规律，然后证明了完全正交滤波器组 PR 条件；在对传统 3 种构建半带滤波器方法说明和实验的基础上设计并实现了基于 ap 的构建半带滤波器的新方法，完成零点分析和最小相位谱分解，整理得到 ap^{1nd}半带滤波器具有完全零点二重性而可进行谱分解得到线性相位响应的结论；最后，分别设计针对一维和二维信号分析和重构实验，并提出基于 ap 半带滤波器的直接分解和综合，具有算法简易的特点且可以方便地设计 M 分辨率谱分析以及基于各种正交基的 ap 半带滤波器。在对图像的分析和重构的对比实验中，ap 2QMFB 恢复图像 PSNR 虽然基本优于 ap 半带滤波器，但最大止不到 2dB，而且也有低于 ap 半带滤波器的情形，这与图像谱分量分布相关。

对于二维信号的分析和重构，目前主要采取行列不相关方式分别进行。这样对于斜方向上谱分量必然造成损失而影响分解和综合。因此，建立一种基于二维传输特性的 ap 滤波器是值得研究的方向。

第 8 章 设计模板的全相位方法

在一维信号处理中，用来抑制噪声干扰并保留原有有用信号的滤波方法以及获得各种信号变换的数字滤波器，有着极其重要和广泛的应用。同样，对于二维和多维信号的处理，各种二维数字滤波方法和信号变换的二维数字滤波器在实际中也有着许多重要的作用。如果将具有理想频率响应 $H_d(u, v)$ 的二维滤波器的冲击响应函数 $h_d(\alpha, \beta)$ 在抽样间隔分别为 $\Delta\alpha$ 和 $\Delta\beta$ 的离散点上进行抽样，则可用所得的理想样本值 $h_d(m, n)$ 的函数：

$$h(\alpha, \beta) = h(m\Delta\alpha, n\Delta\beta) \sum_{m} \sum_{n} h_d(m, n)\delta(\alpha - \Delta\alpha, \beta - \Delta\beta) \tag{8-1}$$

来近似地代表原来的 $h_d(\alpha, \beta)$。其中的 α 和 β 应为离散变量，离散冲激响应与输入离散信号进行滤波的数字运算，相应的滤波器是二维数字滤波器，在二维数字信号处理如图像中应用十分广泛。

按照奈奎斯特取样定律取样并编码得到的数字信号具有较大的数据量，在数字存储及通信中常常考虑如何压缩图像以提高存储效率及节省带宽。最直接的方式是在原图像矩阵的基础上进行再抽取，这种抽取可以在行和/或列向上隔 M 点抽取来实现，这种在经过奈奎斯特取样得到的数字信号基础上再取样的过程称为亚奈奎斯特取样（简称亚奈取样）。图 8-1 显示了几种 $1/K$ 二维亚奈取样结构。

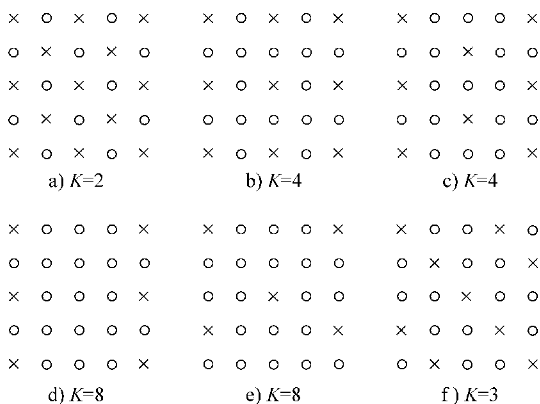


图 8-1　二维亚奈取样结构示例

在节省数据量的同时需要考虑的问题是如何重建亚奈信号，主要基于以下几点：一是利用各种摄像设备获取的自然界图像有水平和垂直两个清晰度，但由于地球重力的影响，实际图像或是要求高的水平清晰度（即垂直细条组成的高频线），或是要求高的垂直清晰度（即水平细条组成的高频线），很少有图像要求水平和垂直清晰度都是高的（相当于噪声）。所以，统计结果表明二维图像信号谱能量 90% 以上集中在二维低频区域，如果进行 2 ∶ 1 或更高亚取样，谱也未发生重叠。二是图像主观质量不完全取决于信噪比，与人眼观察具有低通特性、对色度敏感度低于亮度等因素都有关系，故在确保信噪比高于主观察觉门限值之上就可进行高亚奈取样。

图像的内插处理中，常将图像分解成许多子图像，这种有限的正交变换处理方法由于只能反映图像在局部的频谱分布，因此在块与块的交界处或图像边缘处会引起图像质量的损伤。为了克服边缘效应，一种有效的方法是对图像进行重叠滤波，在重叠滤波中的子图像在行列方向上逐点进行重叠。如果子图像的尺寸为 $N\times N$，则对任一取样点，与该点有关的互相重叠的子图像为 N^2 个，这 N^2 个子图像的输出之和的平均即为重叠滤波的输出。重叠滤波最终可表示为一个 $(2N-1)\times(2N-1)$ 的模板，模板和原始图像做一般卷积即可得处理后的结果。它基本消除了边缘效应，图像的其他性质也有明显的改善。

8.1 模板设计理论

在第 7 章中分析并实现了二维信号的抽取和内插，信号处理过程可用图 8-2 表示。

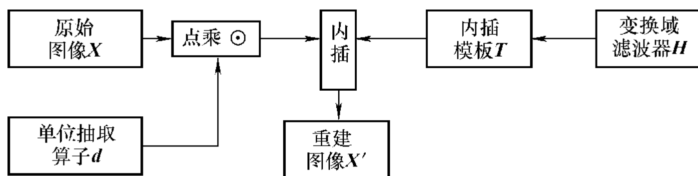


图 8-2 图像的亚奈取样和基于模板的重建

图 8-2 表示原始图像 $\boldsymbol{X}$ 经过单位抽取算子 $\boldsymbol{d}$ 点乘后得到抽取图像 $\boldsymbol{A}$，通过选定某变换域并定义滤波器 $\boldsymbol{H}$ 来建立内插模板 $\boldsymbol{T}$，由 $\boldsymbol{T}$ 和 $\boldsymbol{D}$ 做卷积就可重建信号 $\boldsymbol{X}$。假设输入图像大小为 $N\times N$，则在某正交变换对 $\boldsymbol{\alpha}$ 和 $\boldsymbol{\beta}$ 下 $1/K$ 内插模板设计过程如下：

1）写出 $1/K$ 单位抽取算子 $\boldsymbol{d}$，$\boldsymbol{d}$ 中只有 N^2/K 个元素为 1 而其余为 0。求出 $\boldsymbol{d}$ 的调制点矩阵 $\boldsymbol{D}$，该矩阵只有 $\boldsymbol{K}$ 个元素为 1 而其余为 0。$\boldsymbol{D}$ 与 $\boldsymbol{d}$ 互为变换

对，即

$$D=\frac{1}{N^2}\boldsymbol{\alpha}^{\mathrm{T}}\cdot\boldsymbol{d}\cdot\boldsymbol{\alpha}$$
$$\boldsymbol{d}=\boldsymbol{\beta}^{\mathrm{T}}\cdot\boldsymbol{D}\cdot\boldsymbol{\beta} \tag{8-2}$$

2）写出 $1/K$ 抽取图像 $\boldsymbol{A}$。$\boldsymbol{A}$ 中只有 N^2/K 个图像元素保留而其余均为 0。

$$\boldsymbol{A}=\boldsymbol{X}\odot\boldsymbol{d} \tag{8-3}$$

$\boldsymbol{A}$ 的变换域谱 $\boldsymbol{P}_{\mathrm{A}}$ 中由 K 个 $\boldsymbol{X}$ 谱组成，这 K 个 $\boldsymbol{A}$ 谱应没有重叠部分，否则会产生混叠误差。由正交变换域规律，在时域中相乘对应变换域卷积有

$$\boldsymbol{P}_{\mathrm{A}}=\boldsymbol{X}\otimes\boldsymbol{D} \tag{8-4}$$

3）设计出从 $\boldsymbol{P}_{\mathrm{A}}$ 谱中滤出 $\boldsymbol{X}$ 谱的滤波器 $\boldsymbol{H}$，应符合以下条件：

$$\boldsymbol{P}_{\mathrm{X}}=\boldsymbol{H}\odot\boldsymbol{P}_{\mathrm{A}} \tag{8-5}$$

4）$\boldsymbol{H}$ 的逆变换即是非重叠内插模板 $\boldsymbol{T}$。内插重构图像 $\boldsymbol{X}'$ 可由 $\boldsymbol{T}$ 和 $\boldsymbol{A}$ 卷积得到，即

$$\boldsymbol{X}'=\boldsymbol{T}\otimes\boldsymbol{A} \tag{8-6}$$

上面步骤中最关键的是设计滤波器 $\boldsymbol{H}$，显然 $\boldsymbol{H}$ 是低通滤波器，且为了保证 $\boldsymbol{A}$ 中的保留数据经过式（8-6）处理后不发生改变，模板 $\boldsymbol{T}$ 应满足一定的条件。假设 $\boldsymbol{X}$ 是全 1 的矩阵，结合式（8-3）和式（8-6）得到：

$$\boldsymbol{X}=\boldsymbol{T}\otimes\boldsymbol{d}=>\boldsymbol{P}_{\mathrm{X}}=\boldsymbol{H}\odot\boldsymbol{D} \tag{8-7}$$

直流矩阵 $\boldsymbol{X}$ 的变换域谱 $\boldsymbol{P}_{\mathrm{X}}$ 在（0，0）处有唯一非零值 1。此外，模板 $\boldsymbol{T}$ 与抽取算子 $\boldsymbol{d}$ 的取样处的点乘应等于 1 而其余位置为 0（$\boldsymbol{O}$ 是全 1 矩阵，$\boldsymbol{Z}$ 是 $\boldsymbol{O}$ 的反变换），即

$$\boldsymbol{Z}=\boldsymbol{T}\odot\boldsymbol{d}=>\boldsymbol{O}=\boldsymbol{H}\otimes\boldsymbol{D} \tag{8-8}$$

基于不同的正交变换对应不同的滤波器，变换域中的数值含义也不相同，但它们都从不同的基函数说明了信号的频率分布。在众多正交变换中，最主要的是傅里叶、沃尔什和 DCT 变换。离散沃尔什变换简单、直观，其正交基只有两种状态，用它来表示图像的物理性质较为清楚，因此用沃尔什变换设计的各种图像处理模板系数都是有理数，且规则性强、易于归纳总结。DCT 变换避免了复数运算且能直观反映出能量分布规律，在 MPEG-X 等压缩算法中常用。用离散傅里叶变换设计的模板系数是无理数，但它却有明确的物理意义。下面分别选定二维 Walsh、傅里叶和 DCT 正交变换域，构造 1/2、1/3、1/4 和 1/8 内插模板。

8.1.1　基于二维沃尔什变换内插模板

模板的设计与抽取算子维度大小有关，1/2 抽取算子维度定义为 2×2 和 4×4 两种以作对比，更高维度的推导过程与此基本相同。

1. 1/2 内插模板

1）维度等于 2 时的 1/2 抽取方式，其对应的抽取算子 $\boldsymbol{d}_{1/2}$ 和调制点矩阵 $\boldsymbol{D}_{1/2}$（其中 $\boldsymbol{D}_{1/2}$ 等于 $\boldsymbol{d}_{1/2}$ 的二维沃尔什变换）如下：

$$\boldsymbol{d}_{1/2}=\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\quad \boldsymbol{D}_{1/2}=\begin{bmatrix}2 & 0\\ 0 & 2\end{bmatrix} \tag{8-9}$$

为满足式（8-7）和式（8-8）的要求，求出 $\boldsymbol{H}_{1/2}$ 的通解如下：

$$\boldsymbol{H}_{\mathrm{u}}=\begin{bmatrix}1-z & 1-z_1\\ z_1 & z\end{bmatrix} \tag{8-10}$$

按照通式来设计模板时需要考虑的几点有：首先，如果没有行和列上的高低频分布的先验信息，$\boldsymbol{H}$ 在行列上应对称设置；其次，从上面模板设计过程可知 $\boldsymbol{H}$ 应为低通滤波器，所以选择 $\boldsymbol{H}(0,0)=1$ 且副对角线以下应全部为 0。因此，可以实现的较合适的滤波器（其反变换即为非重叠模板 $\boldsymbol{T}_{1/2}$）如下：

$$\boldsymbol{H}_{1/2}=\begin{bmatrix}1 & 1/2\\ 1/2 & 0\end{bmatrix}\quad \boldsymbol{T}_{1/2}=\begin{bmatrix}0.5 & 0.25\\ 0.25 & 0\end{bmatrix} \tag{8-11}$$

2）维度等于 4 时的 1/2 抽取方式，其对应的抽取算子 $\boldsymbol{d}_{1/2}$ 和调制点矩阵 $\boldsymbol{D}_{1/2}$（其中 $\boldsymbol{D}_{1/2}$ 等于 $\boldsymbol{d}_{1/2}$ 的二维沃尔什变换）如下：

$$\boldsymbol{d}_{1/2}=\begin{bmatrix}1 & 0 & 1 & 0\\ 0 & 1 & 0 & 1\\ 1 & 0 & 1 & 0\\ 0 & 1 & 0 & 1\end{bmatrix}\quad \boldsymbol{D}_{1/2}=\begin{bmatrix}8 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 8\end{bmatrix} \tag{8-12}$$

求出 $\boldsymbol{H}_{1/2}$ 的通解如下：

$$\boldsymbol{H}_{\mathrm{u}}=\begin{bmatrix}1-z & 1-z_1 & 1-z_2 & 1-z_3\\ 1-z_4 & 1-z_5 & 1-z_6 & 1-z_7\\ z_7 & z_6 & z_5 & z_4\\ z_3 & z_2 & z_1 & z\end{bmatrix} \tag{8-13}$$

按上面选定基准定制滤波器 $\boldsymbol{H}_{1/2}$（其反变换即为非重叠模板 $\boldsymbol{T}_{1/2}$）如下：

$$\boldsymbol{H}_{1/2}=\begin{bmatrix}1 & 1 & 1 & 1/2\\ 1 & 1 & 1/2 & 0\\ 1 & 1/2 & 0 & 0\\ 1/2 & 0 & 0 & 0\end{bmatrix}\quad \boldsymbol{T}_{1/2}=\begin{bmatrix}0.5 & 0.25 & 0 & 0.125\\ 0.25 & 0 & -0.125 & 0\\ 0 & -0.125 & 0 & 0\\ 0.125 & 0 & 0 & 0\end{bmatrix} \tag{8-14}$$

比较式（8-11）和式（8-14）可以看出，相同抽取系数 K 定义的不同维数抽取算子 $\boldsymbol{d}$ 具有膨胀特性，即 4 维的抽取算子 ${\boldsymbol{d}_{1/2}}^4$ 可通过膨胀 ${\boldsymbol{d}_{1/2}}^2$ 生成，如下所示：

$$\begin{bmatrix} \times & \times & \times & \times \\ \times & 1 & 1/2 & 0 \\ \times & 1/2 & 0 & 0 \\ \times & 0 & 0 & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 1 & 1 & 1/2 \\ 1 & 1 & 1/2 & 0 \\ 1 & 1/2 & 0 & 0 \\ 1/2 & 0 & 0 & 0 \end{bmatrix} \tag{8-15}$$

2. 1/4 内插模板

1）第一种类型（对应图 8-1b），抽取算子 $\boldsymbol{d}_{1/4}$ 和调制点矩阵 $\boldsymbol{D}_{1/4}$（其中 $\boldsymbol{D}_{1/4}$ 等于 $\boldsymbol{d}_{1/4}$ 的二维沃尔什变换）如下：

$$\boldsymbol{d}_{1/4} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \boldsymbol{D}_{1/4} = \begin{bmatrix} 4 & 0 & 0 & 4 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 4 \end{bmatrix} \tag{8-16}$$

求出 $\boldsymbol{H}_{1/4}$ 的通解如下：

$$\boldsymbol{H}_{\mathrm{u}} = \begin{bmatrix} 1-z_1-z_2-z & 1-z_4-z_5-z_3 & z_3 & z \\ 1-z_7-z_8-z_6 & 1-z_{11}-z_9-z_{10} & z_9 & z_6 \\ z_7 & z_{10} & z_{11} & z_8 \\ z_1 & z_4 & z_5 & z_2 \end{bmatrix} \tag{8-17}$$

按上面选定基准定制滤波器 $\boldsymbol{H}_{1/4}$（其反变换即为非重叠模板 $\boldsymbol{T}_{1/4}$）如下：

$$\boldsymbol{H}_{1/4} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \boldsymbol{T}_{1/4} = \frac{1}{4}\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{8-18}$$

2）第二种类型（对应图 8-1c），抽取算子 $\boldsymbol{d}_{1/4}$ 和调制点矩阵 $\boldsymbol{D}_{1/4}$（其中 $\boldsymbol{D}_{1/4}$ 等于 $\boldsymbol{d}_{1/4}$ 的二维沃尔什变换）如下：

$$\boldsymbol{d}_{1/4} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \boldsymbol{D}_{1/4} = \begin{bmatrix} 4 & 0 & 0 & 4 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 4 & 4 & 0 \end{bmatrix} \tag{8-19}$$

求出 $\boldsymbol{H}_{1/4}$ 的通解如下：

$$\boldsymbol{H}_{\mathrm{u}} = \begin{bmatrix} 1-z_1-z_2-z & 1-z_4-z_5-z_3 & z_3 & z \\ 1-z_7-z_8-z_6 & 1-z_{11}-z_9-z_{10} & z_9 & z_6 \\ z_{10} & z_7 & z_8 & z_{11} \\ z_4 & z_1 & z_2 & z_5 \end{bmatrix} \tag{8-20}$$

定制滤波器 $\boldsymbol{H}_{1/4}$(其反变换即为非重叠模板 $\boldsymbol{T}_{1/4}$) 如下:

$$\boldsymbol{H}_{1/4}=\begin{bmatrix}1 & 1 & 0 & 0\\ 1 & 1/2 & 0 & 0\\ 1/2 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}\quad \boldsymbol{T}_{1/4}=\frac{1}{16}\begin{bmatrix}4 & 4 & 1 & 1\\ 3 & 3 & 0 & 0\\ 0 & 0 & -1 & -1\\ 1 & 1 & 0 & 0\end{bmatrix}\tag{8-21}$$

3. 1/8 内插模板

1) 第一种类型 (对应图 8-1b)。这种类型的算子可以定义为 4 阶, 相应的抽取算子 $\boldsymbol{d}_{1/8}$ 和调制点矩阵 $\boldsymbol{D}_{1/8}$(其中 $\boldsymbol{D}_{1/8}$ 等于 $\boldsymbol{d}_{1/8}$ 的二维沃尔什变换) 如下:

$$\boldsymbol{d}_{1/8}=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 1 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}\quad \boldsymbol{D}_{1/8}=\begin{bmatrix}2 & 2 & 2 & 2\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 2 & 2 & 2 & 2\end{bmatrix}\tag{8-22}$$

求出 $\boldsymbol{H}_{1/8}$ 的通解如下:

$$\boldsymbol{H}_{\mathrm{u}}=\begin{bmatrix}1-\sum\limits_{i=0}^{6}z_i & z_0 & z_1 & z_2\\ 1-\sum\limits_{i=7}^{13}z_i & z_7 & z_8 & z_9\\ z_{10} & z_{11} & z_{12} & z_{13}\\ z_3 & z_4 & z_5 & z_6\end{bmatrix}\tag{8-23}$$

定制滤波器 $\boldsymbol{H}_{1/8}$(其反变换即为非重叠模板 $\boldsymbol{T}_{1/8}$) 如下:

$$\boldsymbol{H}_{1/8}=\begin{bmatrix}1 & 0 & 0 & 0\\ 1 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}\quad \boldsymbol{T}_{1/8}=\frac{1}{16}\begin{bmatrix}2 & 2 & 2 & 2\\ 2 & 2 & 2 & 2\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}\tag{8-24}$$

2) 第二种类型 (对应图 8-1e)。这种类型的算子可以定义为 4 阶, 相应的抽取算子 $\boldsymbol{d}_{1/8}$ 和调制点矩阵 $\boldsymbol{D}_{1/8}$(其中 $\boldsymbol{D}_{1/8}$ 等于 $\boldsymbol{d}_{1/8}$ 的二维沃尔什变换) 如下:

$$\boldsymbol{d}_{1/8}=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}\quad \boldsymbol{D}_{1/8}=\begin{bmatrix}2 & 0 & 2 & 0\\ 0 & 2 & 0 & 2\\ 2 & 0 & 2 & 0\\ 0 & 2 & 0 & 2\end{bmatrix}\tag{8-25}$$

求出 $\boldsymbol{H}_{1/8}$ 的通解如下:

$$H_u = \begin{bmatrix} 1-\sum_{i=0}^{6} z_i & 1-\sum_{i=7}^{13} z_i & z_0 & z_7 \\ z_8 & z_1 & z_9 & z_2 \\ z_{10} & z_3 & z_{11} & z_4 \\ z_5 & z_{12} & z_6 & z_{13} \end{bmatrix} \tag{8-26}$$

定制滤波器 $\boldsymbol{H}_{1/8}$（其反变换即为非重叠模板 $\boldsymbol{T}_{1/8}$）如下：

$$\boldsymbol{H}_{1/8} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \boldsymbol{T}_{1/8} = \frac{1}{16}\begin{bmatrix} 2 & 2 & 0 & 0 \\ 2 & 2 & 0 & 0 \\ 2 & 2 & 0 & 0 \\ 2 & 2 & 0 & 0 \end{bmatrix} \tag{8-27}$$

8.1.2　基于二维傅里叶变换内插模板

正交基为傅里叶算子时，模板设计过程与上面沃尔什算子基本相同，不同之处在于卷积类型变为循环卷积。傅里叶基设计的特点是可以构造任何大小的内插模板，而不限于 2 的整数次幂。下面抽取算子除了 1/3 外，均同沃尔什变换相同。

1. 1/2 内插模板

1）维度等于 2 时的 1/2 抽取方式，调制点 $\boldsymbol{D}_{1/2}$ 等于 $\boldsymbol{d}_{1/2}$ 的二维傅里叶变换，求出 $\boldsymbol{H}_{1/2}$ 的通解如下：

$$\boldsymbol{D}_{1/2} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \quad \boldsymbol{H}_u = \begin{bmatrix} 1-z & 1-z_1 \\ z_1 & z \end{bmatrix} \tag{8-28}$$

与 Walsh 正交基变换相同，定制如下滤波器，其反变换即为非重叠模板 $\boldsymbol{T}_{1/2}$。

$$\boldsymbol{H}_{1/2} = \begin{bmatrix} 1 & 1/2 \\ 1/2 & 0 \end{bmatrix} \quad \boldsymbol{T}_{1/2} = \begin{bmatrix} 0.5 & 0.25 \\ 0.25 & 0 \end{bmatrix} \tag{8-29}$$

2）维度等于 4 时的 1/2 抽取方式，调制点矩阵 $\boldsymbol{D}_{1/4}$ 和滤波器 $\boldsymbol{H}_{1/4}$ 通解如下：

$$\boldsymbol{D}_{1/2} = \begin{bmatrix} 8 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 8 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \boldsymbol{H}_u = \begin{bmatrix} 1-z & 1-z_1 & 1-z_2 & 1-z_3 \\ 1-z_4 & 1-z_5 & 1-z_6 & 1-z_7 \\ z_2 & z_3 & z & z_1 \\ z_6 & z_7 & z_4 & z_5 \end{bmatrix} \tag{8-30}$$

定制如下滤波器，其反变换即为非重叠模板 $\boldsymbol{T}_{1/2}$。

$$\boldsymbol{H}_{1/2} = \begin{bmatrix} 1 & 1 & 1/2 & 1 \\ 1 & 1/2 & 0 & 1/2 \\ 1/2 & 0 & 0 & 0 \\ 1 & 1/2 & 0 & 1/2 \end{bmatrix} \quad \boldsymbol{T}_{1/2} = \frac{1}{16}\begin{bmatrix} 8 & 3 & 0 & 3 \\ 3 & 0 & -1 & 0 \\ 0 & -1 & 0 & -1 \\ 3 & 0 & -1 & 0 \end{bmatrix} \tag{8-31}$$

2. 1/3 内插模板

如图 8-1f 中 1/3 抽取方式的抽取算子 $\boldsymbol{d}_{1/3}$及对应的调制点矩阵 $\boldsymbol{D}_{1/3}$如下：

$$\boldsymbol{d}_{1/3}=\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}\qquad \boldsymbol{D}_{1/3}=\begin{bmatrix}3&0&0\\0&0&3\\0&3&0\end{bmatrix}\tag{8-32}$$

求出 $\boldsymbol{H}_{1/3}$的通解如下：

$$\boldsymbol{H}_{\mathrm{u}}=\begin{bmatrix}1-z_1-z & 1-z_3-z_2 & 1-z_5-z_4\\ z_2 & z_4 & z\\ z_5 & z_1 & z_3\end{bmatrix}\tag{8-33}$$

定制如下滤波器，其反变换即为非重叠模板 $\boldsymbol{T}_{1/3}$。

$$\boldsymbol{H}_{1/3}=\begin{bmatrix}1&1/2&1/2\\1/2&0&0\\1/2&0&0\end{bmatrix}\qquad \boldsymbol{T}_{1/3}=\frac{1}{6}\begin{bmatrix}2&1&1\\1&0&0\\1&0&0\end{bmatrix}\tag{8-34}$$

3. 1/4 内插模板

1）图 8-1b 对应的调制点 $\boldsymbol{D}_{1/4}$及 $\boldsymbol{H}_{1/4}$的通解如下：

$$\boldsymbol{D}_{1/4}=\begin{bmatrix}4&0&4&0\\0&0&0&0\\4&0&4&0\\0&0&0&0\end{bmatrix}\qquad \boldsymbol{H}_{\mathrm{u}}=\begin{bmatrix}1-\sum\limits_{i=0}^{2}z_i & 1-\sum\limits_{i=3}^{5}z_i & z_0 & z_3\\ 1-\sum\limits_{i=6}^{8}z_i & 1-\sum\limits_{i=9}^{11}z_i & z_6 & z_9\\ z_1 & z_4 & z_2 & z_5\\ z_7 & z_{10} & z_8 & z_{11}\end{bmatrix}\tag{8-35}$$

定制如下滤波器，其反变换即为非重叠模板 $\boldsymbol{T}_{1/4}$。

$$\boldsymbol{H}_{1/4}=\begin{bmatrix}1&1/2&0&1/2\\1/2&1/4&0&1/4\\0&0&0&0\\1/2&1/4&0&1/4\end{bmatrix}\qquad \boldsymbol{T}_{1/4}=\frac{1}{16}\begin{bmatrix}4&2&0&2\\2&1&0&1\\0&0&0&0\\2&1&0&1\end{bmatrix}\tag{8-36}$$

2）图 8-1c 对应的调制点 $\boldsymbol{D}_{1/4}$及 $\boldsymbol{H}_{1/4}$的通解如下：

$$\boldsymbol{D}_{1/4}=\begin{bmatrix}4&0&4&0\\0&0&0&0\\0&4&0&4\\0&0&0&0\end{bmatrix}\qquad \boldsymbol{H}_{\mathrm{u}}=\begin{bmatrix}1-\sum\limits_{i=0}^{2}z_i & 1-\sum\limits_{i=3}^{5}z_i & z_0 & z_3\\ 1-\sum\limits_{i=6}^{8}z_i & 1-\sum\limits_{i=9}^{11}z_i & z_6 & z_9\\ z_4 & z_1 & z_5 & z_2\\ z_{10} & z_7 & z_{11} & z_8\end{bmatrix}\tag{8-37}$$

定制如下滤波器，其反变换即为非重叠模板 $\boldsymbol{T}_{1/4}$。

$$\boldsymbol{H}_{1/4}=\begin{bmatrix}1 & 1/2 & 0 & 1/2\\ 1 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 1 & 0 & 0 & 0\end{bmatrix}\quad \boldsymbol{T}_{1/4}=\frac{1}{16}\begin{bmatrix}4 & 3 & 2 & 3\\ 2 & 1 & 0 & 1\\ 0 & -1 & -2 & -1\\ 2 & 1 & 0 & 1\end{bmatrix} \tag{8-38}$$

4. 1/8 内插模板

1）图 8-1d 对应的调制点 $\boldsymbol{D}_{1/8}$ 及 $\boldsymbol{H}_{1/8}$ 的通解如下：

$$\boldsymbol{D}_{1/8}=\begin{bmatrix}2 & 2 & 2 & 2\\ 2 & 2 & 2 & 2\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}\quad \boldsymbol{H}_{\mathrm{u}}=\begin{bmatrix}1-\sum_{i=0}^{6}z_i & z_0 & z_1 & z_2\\ 1-\sum_{i=7}^{13}z_i & z_7 & z_8 & z_9\\ z_3 & z_4 & z_5 & z_6\\ z_{10} & z_{11} & z_{12} & z_{13}\end{bmatrix} \tag{8-39}$$

定制如下滤波器，其反变换即为非重叠模板 $\boldsymbol{T}_{1/8}$。

$$\boldsymbol{H}_{1/8}=\begin{bmatrix}1 & 0 & 0 & 0\\ 1/6 & 1/6 & 0 & 1/6\\ 0 & 0 & 0 & 0\\ 1/6 & 1/6 & 0 & 1/6\end{bmatrix}\quad \boldsymbol{T}_{1/8}=\frac{1}{48}\begin{bmatrix}6 & 4 & 2 & 4\\ 3 & 3 & 3 & 3\\ 0 & 2 & 4 & 2\\ 3 & 3 & 3 & 3\end{bmatrix} \tag{8-40}$$

2）图 8-1e 对应的调制点 $\boldsymbol{D}_{1/8}$ 及 $\boldsymbol{H}_{1/8}$ 的通解如下：

$$\boldsymbol{D}_{1/8}=\begin{bmatrix}2 & 0 & 2 & 0\\ 0 & 2 & 0 & 2\\ 2 & 0 & 2 & 0\\ 0 & 2 & 0 & 2\end{bmatrix}\quad \boldsymbol{H}_{\mathrm{u}}=\begin{bmatrix}1-\sum_{i=0}^{6}z_i & 1-\sum_{i=7}^{13}z_i & z_0 & z_7\\ z_8 & z_1 & z_9 & z_2\\ z_3 & z_{10} & z_4 & z_{11}\\ z_{12} & z_5 & z_{13} & z_6\end{bmatrix} \tag{8-41}$$

定制如下滤波器，其反变换即为非重叠模板 $\boldsymbol{T}_{1/8}$。

$$\boldsymbol{H}_{1/8}=\begin{bmatrix}1 & 1/4 & 0 & 1/4\\ 1/4 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 1/4 & 0 & 0 & 0\end{bmatrix}\quad \boldsymbol{T}_{1/8}=\frac{1}{32}\begin{bmatrix}4 & 3 & 2 & 3\\ 3 & 2 & 1 & 2\\ 2 & 1 & 0 & 1\\ 3 & 2 & 1 & 2\end{bmatrix} \tag{8-42}$$

8.1.3　基于二维 DCT 变换内插模板

由第 5 章内容可知，二维 DCT 运算可以转化成 DFT 运算，然后取结果的实部作为 DCT 变换的结果。从这个意义上理解，如果不考虑相位而只考虑内插数

据的幅值，则正交基为 DCT 算子模板设计过程与傅里叶算子基本相同，不同之处是 $\boldsymbol{H}$ 的选择只要以原点（左上角）对称即可。DCT 基可以设计任何大小的内插模板，而不限于 2 的整数次幂。以上面傅里叶基构造的结果直接得到 DCT 基滤波器 H 和对应的非重叠模板。

1. 1/2 内插模板

1）维度等于 2 时的 1/2 抽取方式，$\boldsymbol{H}_{1/2}$和非重叠模板 $\boldsymbol{T}_{1/2}$与傅里叶基情形相同。

2）维度等于 4 时的 1/2 抽取方式，定制 $\boldsymbol{H}_{1/2}$如下，其反变换即为非重叠模板 $\boldsymbol{T}_{1/2}$。

$$\boldsymbol{H}_{1/2}=\begin{bmatrix}1 & 1 & 1/2 & 1\\ 1 & 1/2 & 0 & 0\\ 1/2 & 0 & 0 & 0\\ 1 & 0 & 0 & 0\end{bmatrix}\quad \boldsymbol{T}_{1/2}=\begin{bmatrix}1.6373 & 0.6090 & 0.8149 & 0.2866\\ 0.6090 & -0.3461 & -0.0366 & -0.4917\\ 0.8149 & -0.0366 & 0.4193 & 0.0678\\ 0.2866 & -0.4917 & 0.0678 & -0.2105\end{bmatrix} \tag{8-43}$$

2. 1/4 内插模板

1）图 8-1b 对应的定制 $\boldsymbol{H}_{1/4}$如下，其反变换即为非重叠模板 $\boldsymbol{T}_{1/4}$。

$$\boldsymbol{H}_{1/4}=\begin{bmatrix}1 & 1/2 & 0 & 1/2\\ 1/2 & 1/4 & 0 & 0\\ 0 & 0 & 0 & 0\\ 1/2 & 0 & 0 & 0\end{bmatrix}\quad \boldsymbol{T}_{1/4}=\begin{bmatrix}0.8186 & 0.4295 & 0.5324 & 0.1433\\ 0.4295 & 0.0770 & 0.2317 & -0.1208\\ 0.5324 & 0.2317 & 0.4596 & 0.1589\\ 0.1433 & -0.1208 & 0.1589 & -0.1052\end{bmatrix} \tag{8-44}$$

2）图 8-1c 对应的定制 $\boldsymbol{H}_{1/4}$如下，其反变换即为非重叠模板 $\boldsymbol{T}_{1/4}$。

$$\boldsymbol{H}_{1/4}=\begin{bmatrix}1 & 1/2 & 0 & 1/2\\ 1/2 & 1/4 & 0 & 0\\ 0 & 0 & 0 & 0\\ 1/2 & 0 & 0 & 0\end{bmatrix}\quad \boldsymbol{T}_{1/4}=\begin{bmatrix}0.8186 & 0.4295 & 0.5324 & 0.1433\\ 0.4295 & 0.0770 & 0.2317 & -0.1208\\ 0.5324 & 0.2317 & 0.4596 & 0.1589\\ 0.1433 & -0.1208 & 0.1589 & -0.1052\end{bmatrix} \tag{8-45}$$

3. 1/8 内插模板

1）图 8-1d 对应的定制 $\boldsymbol{H}_{1/8}$如下，其反变换即为非重叠模板 $\boldsymbol{T}_{1/8}$。

$$\boldsymbol{H}_{1/8}=\begin{bmatrix}1 & 0 & 0 & 0\\ 1/6 & 1/6 & 0 & 0\\ 0 & 0 & 0 & 0\\ 1/6 & 0 & 0 & 0\end{bmatrix}\quad \boldsymbol{T}_{1/8}=\begin{bmatrix}0.3981 & 0.3565 & 0.2975 & 0.2559\\ 0.2476 & 02303 & 02059 & 0.1886\\ 0.2524 & 0.2697 & 0.2941 & 0.3114\\ 0.1019 & 0.1435 & 0.2025 & 0.2441\end{bmatrix} \tag{8-46}$$

2）图 8-1e 对应的定制 $\boldsymbol{H}_{1/8}$ 如下，其反变换即为非重叠模板 $\boldsymbol{T}_{1/8}$。

$$\boldsymbol{H}_{1/8}=\begin{bmatrix}1 & 1/4 & 0 & 1/4\\ 1/4 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 1/4 & 0 & 0 & 0\end{bmatrix}\quad \boldsymbol{T}_{1/8}=\begin{bmatrix}0.4810 & 0.3176 & 0.4133 & 0.2400\\ 0.3176 & 0.1543 & 0.2500 & 0.0867\\ 0.4133 & 0.2500 & 0.3457 & 0.1824\\ 0.2500 & 0.0867 & 0.1824 & 0.0190\end{bmatrix} \tag{8-47}$$

4. 1/3 内插模板

图 8-1c 对应的定制 $\boldsymbol{H}_{1/3}$ 如下，其反变换即为非重叠模板 $\boldsymbol{T}_{1/3}$。

$$\boldsymbol{H}_{1/3}=\begin{bmatrix}1 & 1/2 & 1/2\\ 1/2 & 0 & 0\\ 1/2 & 0 & 0\end{bmatrix}\quad \boldsymbol{T}_{1/3}=\begin{bmatrix}0.9773 & 0.4196 & 0.5690\\ 0.4196 & -0.1381 & 0.0114\\ 0.5690 & 0.0114 & 0.1608\end{bmatrix} \tag{8-48}$$

实现以上结果的 MATLAB 代码（只列出傅里叶基）如下：

```
function out = fft2T

clc,path(path,'d:\work\book1\chapter 5');

if 1
    d = decimator(82);N = size(d,1);
    D = fft2(d,N,N),M = max(max(D));H = sloveH(D/M)
else
    H = input('Please input transmission property H \n');
    N = size(H,1);T = ifft2(H,N,N)
    if nargout ~ =0
        out = T;
    end
end
function H = sloveH(D)
D = int8(real(fliplr(flipud(D))));[R,C] = size(D);
id = num2str(R);HX = decimator(id);
rx = [1:R];cx = [1:C];
for r =1:R
    rxx = mod(rx - r,R) +1;rxx(rxx = =0) = R;
    for c =1:C
        cxx = mod(cx - c,C) +1;cxx(cxx = =0) = C;
        for rr =1:R
            for cc =1:C
                dx(rr,cc) = D(rxx(rr),cxx(cc));
            end
        end
```

```
        eqs((r-1)*R+c)=sum(sum(HX.*dx))-1;
    end
end
switch id
    case '2'
        [x0 x1 y0 y1]=solve(eqs);H=[x0 x1;y0 y1];
    case '3'
        [x0 x1 x2 y0 y1 y2 u0 u1 u2]=solve(eqs);
        H=[x0 x1 x2;y0 y1 y2;u0 u1 u2];
    case '4'
        [x0 x1 x2 x3 y0 y1 y2 y3 u0 u1 u2 u3 v0 v1 v2 v3]=solve(eqs);
        H=[x0 x1 x2 x3;y0 y1 y2 y3;u0 u1 u2 u3;v0 v1 v2 v3];
    otherwise
        error('Invalid Parameter');
end
```

程序中函数 decimator 用来根据输入的抽取类型生成各种抽取算子，并构造符号矩阵 HX 以求出传输特性 $\boldsymbol{H}$ 的通解。

8.2 亚奈取样图像的二维谱分析

选取不同的正交基对二维图像进行正交变换可得到变换域的矩阵形式，由第 5 章内容已知，变换域矩阵中各系数就是正交基图像的加权，而正交基图像体现了空间亮度变换的快慢即具有空间频率的概念。把变换域矩阵称为原图像的二维谱，不同的正交变换算子得到的基图像不同，对应的二维谱含义也不相同。一般把沃尔什变换得到的谱成为列率谱；把傅里叶变换和 DCT 得到的谱称为频谱，DCT 对应实数谱即不含相位信息，傅里叶则对应复数谱即包含相位信息。本节从二维沃尔什列率谱和傅里叶频谱两方面分析亚奈取样图像前后发生的变换，从而得到设计滤波器的基本思想。

8.2.1 二维列率谱分析

一幅按二维取样定理所得图像数据可用二维矩阵表示，在按传统方式处理时，图像被分成某固定大小的子图像，这里假设子图像 $\boldsymbol{a}$ 大小是 4×4，表示成如下：

$$\boldsymbol{a}=\begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix} \tag{8-49}$$

图像 $\boldsymbol{a}$ 经过二维沃尔什变换得到二维列率谱矩阵 $\boldsymbol{A}$。图像的谱能量主要集中在二维低列率区域，高列率能量较少。由沃尔什基图像构成来看，列率谱从 $\boldsymbol{A}$ 左上角向右向下行和列的列率逐渐增大，行列双向最大列率谱出现在 $\boldsymbol{A}$ 的右下角。若忽略高列率谱成分后 $\boldsymbol{A}$ 可表示为

$$\boldsymbol{A}=\begin{bmatrix} s_{00} & s_{01} & s_{02} & 0 \\ s_{10} & s_{11} & 0 & 0 \\ s_{20} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{8-50}$$

在上面介绍的各种抽取算子中 1/2 是最常用的一种方式，对应的抽取矩阵有两种，第一种 $\boldsymbol{d}_1$ 与列率谱 $\boldsymbol{D}_1$ 如下：

$$\boldsymbol{d}_1=\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \qquad \boldsymbol{D}_1=\frac{1}{2}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8-51}$$

显然，

$$\boldsymbol{d}_1=[\boldsymbol{W}(0,0)+\boldsymbol{W}(3,3)]/2 \tag{8-52}$$

式（8-52）中的 $\boldsymbol{W}$ 是二维沃尔什基图像。图像 $\boldsymbol{a}$ 经过 $\boldsymbol{d}_1$ 抽取后得到 $\boldsymbol{a}_{\mathrm{d1}}$：

$$\boldsymbol{a}_{\mathrm{d1}}=\boldsymbol{a}\odot\boldsymbol{d}_1=\boldsymbol{a}\odot[\boldsymbol{W}(0,0)+\boldsymbol{W}(3,3)]/2=(\boldsymbol{a}+\boldsymbol{b}_1)/2=\begin{bmatrix} a_{00} & 0 & a_{02} & 0 \\ 0 & a_{11} & 0 & a_{13} \\ a_{20} & 0 & a_{22} & 0 \\ 0 & a_{31} & 0 & a_{33} \end{bmatrix} \tag{8-53}$$

其中，图像 $\boldsymbol{b}_1$ 是输入矩阵 $\boldsymbol{a}$ 与 $\boldsymbol{W}(3,3)$ 的点乘，其列率谱 $\boldsymbol{B}_1$ 等于 $\boldsymbol{A}$ 与 $\boldsymbol{W}(3,3)$ 列率谱的并元卷积。由于 $\boldsymbol{W}(3,3)$ 是沃尔什基图像，所以 $\boldsymbol{B}_1$ 的成分与 $\boldsymbol{A}$ 相同，元素位置则都与（3，3）进行了并元移位，由式（8-53）可知谱 $\boldsymbol{A}_{\mathrm{d1}}$ 等于 $\boldsymbol{A}$ 和 $\boldsymbol{B}_1$ 之和（省略 1/2）。

$$\boldsymbol{A}_{\mathrm{d1}}=\boldsymbol{A}+\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & s_{20} \\ 0 & 0 & s_{11} & s_{10} \\ 0 & s_{02} & s_{01} & s_{00} \end{bmatrix}=\begin{bmatrix} s_{00} & s_{01} & s_{02} & 0 \\ s_{10} & s_{11} & 0 & s_{20} \\ s_{20} & 0 & s_{11} & s_{10} \\ 0 & s_{02} & s_{01} & s_{00} \end{bmatrix} \tag{8-54}$$

可见原图像 $\boldsymbol{A}$ 经过调制矩阵 $\boldsymbol{D}_1$ 后增加了右下角高列率谱成分。

第二种 $\boldsymbol{d}_2$ 与列率谱 $\boldsymbol{D}_2$ 如下：

$$\boldsymbol{d}_2=\begin{bmatrix}0&1&0&1\\1&0&1&0\\0&1&0&1\\1&0&1&0\end{bmatrix}\qquad \boldsymbol{D}_2=\frac{1}{2}\begin{bmatrix}1&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&-1\end{bmatrix} \tag{8-55}$$

比较式（8-55）和式（8-51），可以看到两种调制的区别仅在于高列率谱成分符号，所以第二种调制得到的图像 $\boldsymbol{a}_{\mathrm{d}2}$ 及谱 $\boldsymbol{A}_{\mathrm{d}2}$ 如下：

$$\boldsymbol{a}_{\mathrm{d}2}=\frac{\boldsymbol{a}+\boldsymbol{b}_2}{2}=\begin{bmatrix}0&a_{01}&0&a_{03}\\a_{10}&0&a_{12}&0\\0&a_{21}&0&a_{23}\\a_{30}&0&a_{32}&0\end{bmatrix}\qquad \boldsymbol{A}_{\mathrm{d}2}=\begin{bmatrix}s_{00}&s_{01}&s_{02}&0\\s_{10}&s_{11}&0&-s_{20}\\s_{20}&0&-s_{11}&-s_{10}\\0&-s_{02}&-s_{01}&-s_{00}\end{bmatrix} \tag{8-56}$$

观察式（8-54）和式（8-56），$\boldsymbol{A}_{\mathrm{d}1}$ 和 $\boldsymbol{A}_{\mathrm{d}2}$ 的谱在低列率部分相同且都等于原信号谱，这也说明通过设计某种低通滤波器如式（8-53）对 $\boldsymbol{a}_{\mathrm{d}1}$ 进行重构。

8.2.2 二维频谱分析

对于傅里叶频谱分布，如果不做中心点移动，高频成分在行和列的中心处，只留在左边第一列和上面第一行，低频成分沿着中心行列在上下左右分布。如果仍假定图像 $\boldsymbol{a}$ 只包含中低频，则其频谱 $\boldsymbol{F}_{\mathrm{a}}$ 可表示如下：

$$\boldsymbol{F}_{\mathrm{a}}=\begin{bmatrix}f_{00}&f_{01}&0&f_{03}\\f_{10}&0&0&0\\0&0&0&0\\f_{30}&0&0&0\end{bmatrix} \tag{8-57}$$

经过抽取后的图像与沃尔什变换情形相同，不再列出。对于第一种抽样得到图像 $\boldsymbol{a}_{\mathrm{d}1}$ 的谱 $\boldsymbol{F}_{\mathrm{ad}1}$ 等于原图像频谱 $\boldsymbol{F}_{\mathrm{a}}$ 和调制后图像 $\boldsymbol{b}_1$ 的谱 $\boldsymbol{F}_{\mathrm{b}1}$ 之和（省略 1/2）。

$$\boldsymbol{F}_{\mathrm{ad}1}=\boldsymbol{F}_{\mathrm{a}}+\boldsymbol{F}_{\mathrm{b}1}=\boldsymbol{F}_{\mathrm{a}}+\boldsymbol{F}_{\mathrm{a}}\otimes\mathrm{F}\left\{\begin{bmatrix}1&-1&1&-1\\-1&1&-1&1\\1&-1&1&-1\\-1&1&-1&1\end{bmatrix}\right\}=\begin{bmatrix}f_{00}&f_{01}&0&f_{03}\\f_{10}&&f_{30}&\\&f_{03}&f_{00}&f_{01}\\f_{30}&&f_{10}&\end{bmatrix} \tag{8-58}$$

同理可得第二类 1/2 亚奈抽样信号的频谱成分相同，只是高频符号相反。所以两种抽样得到的信号都保留了原有信号频谱，所以可以采用如式（8-31）所示低通滤波器处理恢复原信号。

这种通过低通滤波器进行图像还原的方法具有算法简单的优点，但是子图像内插滤波恢复得到的信号质量较差，在斜线处会出现锯齿效应。这主要是因为图像由子块拼合而成，在子块交界处存在的丰富高频成分被忽略而出现边缘效应；

其二，图像频谱虽然大部分集中在二维低频区，但在二维高频区仍占有一定的比例，在调制过程中出现混叠，在使用低通滤波器进行重建时而出现混叠失真。下面分析含有高频成分的图像在进行 1/2 亚取样产生混叠的原因，影响大小及消除混叠的基本思想。

存在高频成分的二维图像 $\boldsymbol{a}$ 的频率表示如下：

$$\boldsymbol{A}=\begin{bmatrix} f_{00} & f_{01} & 0 & f_{03} \\ f_{10} & 0 & 0 & 0 \\ 0 & 0 & f_{22} & 0 \\ f_{30} & 0 & 0 & 0 \end{bmatrix} \tag{8-59}$$

两种方式 1/2 抽取图像信号谱如下：

$$\boldsymbol{A}_{\mathrm{d1}}=\begin{bmatrix} f_{00}+f_{22} & f_{01} & 0 & f_{03} \\ f_{10} & 0 & f_{30} & 0 \\ 0 & f_{03} & f_{22}+f_{00} & f_{01} \\ f_{30} & 0 & f_{10} & 0 \end{bmatrix} \quad \boldsymbol{A}_{\mathrm{d2}}=\begin{bmatrix} f_{00}-f_{22} & f_{01} & 0 & f_{03} \\ f_{10} & 0 & -f_{30} & 0 \\ 0 & -f_{03} & f_{22}-f_{00} & -f_{01} \\ f_{30} & 0 & -f_{10} & 0 \end{bmatrix} \tag{8-60}$$

选定低通滤波器过滤 $\boldsymbol{A}_{\mathrm{d1}}$ 得到的图像中引入了高频分量干扰，而过滤 $\boldsymbol{A}_{\mathrm{d2}}$ 得到的图像中引入的高频干扰具有相反的相位，如果两者结果相加平均就可以完全消除高频干扰。这两种抽取方式的置零方式只差一个像素，但其频谱性质中高频分量相位相反。利用此性质构造重叠滤波方式就可消除边缘效应，改善内插重构图像的质量。

8.3　全相位内插模板设计

全相位模板设计的基本思想是建立在重叠处理方法基础之上的，但这种重叠是相关输入 $\boldsymbol{a}$ 的全部邻接移动矩阵即最大程度重叠。按照上面分析结果，子图像中高频成分相位互反，因而通过平均结果的方式求出最终内插数据。分别以沃尔什变换基和傅里叶基变换介绍全相位内插模板设计过程。

8.3.1　沃尔什内插模板

按照 4×4 分割的子图像窗口在行列方向上移动产生相互重叠的其他位置子图像，与其中某像素 P 有关的共有 16 个，遍及在原始图像中 7×7 像素的范围内。

如图 8-3 所示，图中没有把与像素 P 相关的 16 个子图全部画出，只列出了 4 个子图像。对与像素 P 相关即包含其在内的 16 个子图像经过二维低通滤波器处

理后的输出值平均值（这种平均可以是加权平均）即为重叠滤波器的输出。显然，采用这种重叠滤波器内插就可以消除高列率谱的混叠。如图 8-3 所示，对于 1/2 取样图像，相邻的子图像反复出现 $\boldsymbol{h}_1$ 和 $\boldsymbol{h}_2$ 结构，这两种结构的高列率分量幅度近似相等而相位相反，所以在重叠相加时基本上对消，从而有效地克服了按子图分块滤波恢复拼接时的边缘效应。

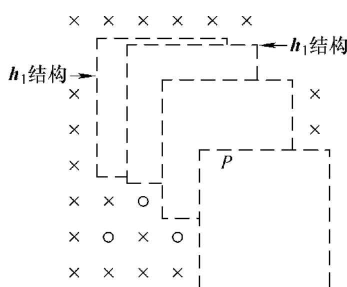


图 8-3　4 ×4 子图像重叠滤波的构成

下面讨论最简单的子图像为 2 ×2 的二维低通重叠滤波器。如图 8-4 所示，与像素 $\boldsymbol{a}_5$ 相关的图像共有 4 个子块 $\boldsymbol{X}_i$，其包含的像素分别为［$\boldsymbol{a}_i$，$\boldsymbol{a}_{i+1}$；$\boldsymbol{a}_{i+3}$，$\boldsymbol{a}_{i+4}$］（i =1，2，4，5）。

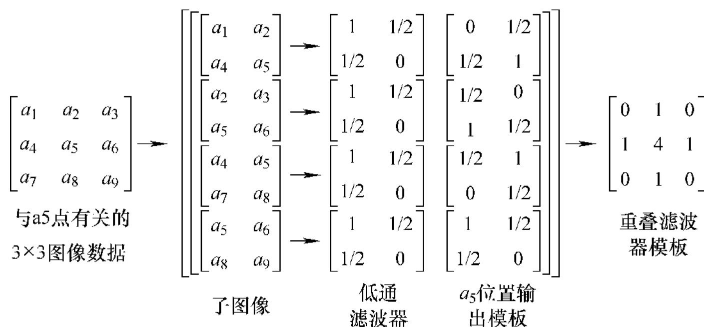


图 8-4　对应 2 阶低通列率滤波器的 3 阶重叠模板构造

某个 2 ×2 的子图像 $\boldsymbol{a}$ 经过二维沃尔什变换后为

$$\begin{aligned}\boldsymbol{S}_{\mathrm{a}} &= \frac{1}{2^2}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\begin{bmatrix}\boldsymbol{a}_i & \boldsymbol{a}_{i+1}\\\boldsymbol{a}_{i+3} & \boldsymbol{a}_{i+4}\end{bmatrix}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\\&= \frac{1}{4}\begin{bmatrix}\boldsymbol{a}_i+\boldsymbol{a}_{i+1}+\boldsymbol{a}_{i+3}+\boldsymbol{a}_{i+4} & \boldsymbol{a}_i-\boldsymbol{a}_{i+1}+\boldsymbol{a}_{i+3}-\boldsymbol{a}_{i+4}\\\boldsymbol{a}_i+\boldsymbol{a}_{i+1}-\boldsymbol{a}_{i+3}-\boldsymbol{a}_{i+4} & \boldsymbol{a}_i-\boldsymbol{a}_{i+1}-\boldsymbol{a}_{i+3}+\boldsymbol{a}_{i+4}\end{bmatrix}\end{aligned} \tag{8-61}$$

经过滤波器 $\boldsymbol{H}_2$ 后变为

$$\boldsymbol{S}'_{\mathrm{a}} = \boldsymbol{S}_{\mathrm{a}} \odot \boldsymbol{H}_2 = \frac{1}{8}\begin{bmatrix}2(\boldsymbol{a}_i+\boldsymbol{a}_{i+1}+\boldsymbol{a}_{i+3}+\boldsymbol{a}_{i+4}) & \boldsymbol{a}_i-\boldsymbol{a}_{i+1}+\boldsymbol{a}_{i+3}-\boldsymbol{a}_{i+4}\\\boldsymbol{a}_i+\boldsymbol{a}_{i+1}-\boldsymbol{a}_{i+3}-\boldsymbol{a}_{i+4} & 0\end{bmatrix} \tag{8-62}$$

再经过二维沃尔什反变换得到内插恢复的子图像 $\boldsymbol{a}'$ 如下：

$$\boldsymbol{X}' = \begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\boldsymbol{S}'_{\mathrm{a}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix} = \frac{1}{2}\begin{bmatrix}2\boldsymbol{a}_i+\boldsymbol{a}_{i+1}+\boldsymbol{a}_{i+3} & \boldsymbol{a}_i+2\boldsymbol{a}_{i+1}+\boldsymbol{a}_{i+4}\\\boldsymbol{a}_i+2\boldsymbol{a}_{i+3}+\boldsymbol{a}_{i+4} & \boldsymbol{a}_{i+1}+\boldsymbol{a}_{i+3}+2\boldsymbol{a}_{i+4}\end{bmatrix} \tag{8-63}$$

当 i =1 ~4，对应点 $\boldsymbol{a}_5$ 的输出分别如下：

$$[\boldsymbol{b}_5]_1 = \boldsymbol{a}_5 + \boldsymbol{a}_4/2 + \boldsymbol{a}_2/2 \tag{8-64}$$

$$[\boldsymbol{b}_5]_2 = \boldsymbol{a}_5 + \boldsymbol{a}_2/2 + \boldsymbol{a}_6/2 \tag{8-65}$$

$$[\boldsymbol{b}_5]_3 = \boldsymbol{a}_5 + \boldsymbol{a}_4/2 + \boldsymbol{a}_8/2 \tag{8-66}$$

$$[\boldsymbol{b}_5]_4 = \boldsymbol{a}_5 + \boldsymbol{a}_6/2 + \boldsymbol{a}_8/2 \tag{8-67}$$

上面 4 个表达式（8-64）~式(8-67）的输出和的平均就是重叠滤波器输出 $\boldsymbol{b}_5$，即

$$\boldsymbol{b}_5 = \frac{1}{4}\sum_{i=1}^{4}[\boldsymbol{b}_5]_i = \boldsymbol{a}_5 + \frac{1}{4}\sum_{i=1}^{4}\boldsymbol{a}_{2i} \tag{8-68}$$

式（8-68）可用一个模板来表示为

$$\boldsymbol{T}_3 = \frac{1}{4}\begin{bmatrix} 0 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \tag{8-69}$$

模板与图像做尺寸为 3×3 的卷积即得重叠滤波的全部输出，其结果与全部 2×2 子图像逐行逐点移动作低通滤波器后的叠加输出平均相同。这种基于模板的内插方法计算简单，对于 1/2 抽取后保留的像素内插后处理不变，对于内插产生的新像素值等于周围 4 个像素的平均值。

以上分析的子图像尺寸是 2×2，当图像尺寸为 $N \times N$ 时，与某样点相关的子图像有 N^2 个，它们分别滤波后并进行叠加的平均输出，可以表示成一个尺寸大小为 $(2N-1)\times(2N-1)$ 的模板，这个模板与图像的逐点卷积即得重叠滤波的全部输出。显然，这就是我们在第 5 章中设计的二维全相位滤波器或称全相位内插模板。

全相位沃尔什内插模板的设计就是给定低通滤波器 $\boldsymbol{H}$ 求出模板 $\boldsymbol{T}$ 的过程。以一维 N 阶沃尔什变换域列率滤波器为例，若输入数据为 $x(n)$，列率滤波器的变换域特性为 $H(k)$，滤波后输出应为 $\boldsymbol{x}$ 与 $\boldsymbol{h}$ 的并元卷积。每一子段经过 $\boldsymbol{H}$ 滤波的输出重叠过程如图 8-5 所示。

数据x	A		B		C		D		E		F		G
第一段输出							$Dh(0)$	+	$Eh(1)$	+	$Eh(2)$	+	$Gh(3)$
第二段输出					$Ch(1)$	+	$Dh(0)$	+	$Eh(3)$	+	$Fh(2)$		
第三段输出			$Bh(2)$	+	$Ch(3)$	+	$Dh(0)$	+	$Eh(1)$				
第四段输出	$Ah(3)$	+	$Bh(2)$	+	$Ch(1)$	+	$Dh(0)$						
重叠后输出	A	+	B	+	C	+	D	+	E	+	F	+	G
	$h(3)$		$2h(2)$		$2h(1)+h(3)$		$4h(0)$		$2h(1)+h(3)$		$2h(2)$		$h(3)$

图 8-5　4 阶并元移位及重叠

由图 8-5 重叠输出结果可以看出，一维全相位内插模板可以表示成矩阵 $\boldsymbol{Q}$ 与 $\boldsymbol{h}$ 相乘的形式：

$$\boldsymbol{T} = \boldsymbol{Q}\cdot\boldsymbol{h} = \frac{1}{N}\boldsymbol{Q}\cdot\boldsymbol{W}\cdot\boldsymbol{H} \tag{8-70}$$

式（8-70）中，$\boldsymbol{W}$ 是沃尔什基矩阵；$\boldsymbol{Q}$ 是并元移位叠加矩阵，定义如下：

$$\boldsymbol{Q}=\begin{bmatrix}0&0&0&4&0&0&0\\0&0&2&0&2&0&0\\0&2&0&0&0&2&0\\1&0&1&0&1&0&1\end{bmatrix}^{\mathrm{T}} \tag{8-71}$$

总结以上，对于 N 阶二维沃尔什全相位滤波，从传输矩阵 $\boldsymbol{H}$ 求出模板 $\boldsymbol{T}$ 的过程如下：

$$\boldsymbol{T}=\frac{1}{N^2}\boldsymbol{Q}\cdot\boldsymbol{W}\cdot\boldsymbol{H}\cdot\boldsymbol{W}\cdot\boldsymbol{Q}^{\mathrm{T}} \tag{8-72}$$

式中，$\boldsymbol{W}$ 是 N 阶沃尔什基矩阵；$\boldsymbol{Q}$ 是 $N\times(2N-1)$ 并元移位叠加矩阵，定义如下：

$$\boldsymbol{Q}=\begin{bmatrix}q_{0,N-1}&q_{1,N-1}&\cdots&q_{N-1,N-1}\\\cdots&\cdots&\cdots&\cdots\\q_{0,1}&q_{1,1}&\cdots&q_{N-1,1}\\q_{0,0}&q_{1,0}&\cdots&q_{N-1,0}\\q_{0,-1}&q_{1,-1}&\cdots&q_{N-1,-1}\\\cdots&\cdots&\cdots&\cdots\\q_{0,-N+1}&q_{1,-N+1}&\cdots&q_{N-1,-N+1}\end{bmatrix}_{(2N-1)\times N}, \tag{8-73}$$

$$N=2^p,\ p\in Z^+$$

$$\begin{cases}q_{0,0}=1 & M=2^k,\ k=0,1,\cdots,p-1;\\ q_{M+i,\pm(M+j)}=\dfrac{q_{i,j}}{2} & i=0,1,\cdots,M-1;j=0,\pm1,\cdots,\pm(M-1)\\ q_{i,j}=0 & \text{其他}\end{cases}$$

根据式（8-72），如果已知模板 $\boldsymbol{T}$，可通过下式求其对应的传输矩阵 $\boldsymbol{H}$：

$$\boldsymbol{H}=\boldsymbol{W}\cdot\boldsymbol{Q}^{\#}\cdot\boldsymbol{T}\cdot(\boldsymbol{Q}^{\#})^{\mathrm{T}}\cdot\boldsymbol{W} \tag{8-74}$$

式中，$\boldsymbol{Q}^{\#}$是并元移位叠加矩阵 $\boldsymbol{Q}$ 的伪逆。

8.3.2 傅里叶内插模板

根据第 5 章全相位 DFT 基的概念，全相位 DFT 基 $\boldsymbol{A}$ 就是内插模板 $\boldsymbol{T}$。即如果已知传输矩阵 $\boldsymbol{H}$，$\boldsymbol{T}$ 可按下式求出：

$$\boldsymbol{T}=\frac{1}{N^2}\boldsymbol{E}\cdot\boldsymbol{F}\cdot\boldsymbol{H}\cdot\boldsymbol{F}^*\cdot\boldsymbol{E}^{\mathrm{T}} \tag{8-75}$$

$\boldsymbol{F}^*$ 是傅里叶基矩阵 $\boldsymbol{F}$ 的共轭。$\boldsymbol{E}$ 是全相移位矩阵，其定义如下：

$$\boldsymbol{E}=\begin{bmatrix} 0 & \cdots & 0 & 1 \\ 0 & \cdots & 2 & 0 \\ \vdots & \iddots & \vdots & \vdots \\ N & \cdots & 0 & 0 \\ 0 & \cdots & 0 & N-1 \\ 0 & \cdots & N-2 & 0 \\ \vdots & \iddots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 \end{bmatrix}_{(2N-1)\times N} \tag{8-76}$$

第 5 章已通过实验证明，通过加不同窗函数可使模板性能得到某种程度的改善。容易证明，全相移位矩阵 $\boldsymbol{E}$ 变成：

$$\boldsymbol{E}=\begin{bmatrix} 0 & \cdots & 0 & \boldsymbol{C}_{N-1} \\ 0 & \cdots & \boldsymbol{C}_{N-2} & 0 \\ \vdots & \iddots & \vdots & \vdots \\ \boldsymbol{C}_0 & \cdots & 0 & 0 \\ 0 & \cdots & 0 & \boldsymbol{C}_{-1} \\ 0 & \cdots & \boldsymbol{C}_{-2} & 0 \\ \vdots & \iddots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 \end{bmatrix}_{(2N-1)\times N} \tag{8-77}$$

式中，$\boldsymbol{C}$ 是卷积窗，式（5-75）中 N 也将变更为 $\boldsymbol{C}_0$。

全相位 DCT 基内插模板可按照傅里叶基内插模板设计过程，不再赘述。整理以上，沃尔什域、傅里叶域和 cosine 域全相位内插模板设计过程如下：

1）根据抽取算子 $\boldsymbol{d}$ 的类型确定阶数 N 并选择适宜传输特性 $\boldsymbol{H}$；

2）选择前后基窗并计算出卷积窗 $\boldsymbol{C}$；

3）按照式（8-73）或式（8-77）计算并元移位叠加矩阵 $\boldsymbol{Q}$ 或全相移位矩阵 $\boldsymbol{E}$；

4）按照式（8-72）或式（8-75）构造出模板 $\boldsymbol{T}$。

8.4　传统插值算法

最早应用的插值算法是线性插值，其具有计算量小，插值速度快等特点。随后 Herman 等人将 B 样条应用于医学图像插值上，但这两种算法插值后使边界变得模糊。自适应插值则在分析图像结构的基础上，不同特征的区域采用不同的插值算法以适应各方向的边缘，但计算量又较大。小波分析是时间和频率的局部变换，能有效地提取出信号信息，被誉为“数字显微镜”。小波插值很好地解决了图像边界过于光滑的缺陷，通过保留各子带中不多系数而使图像得以完美重构，

但算法较复杂且收敛性取决于小波性质，算法不稳定。下面对线性插值、立方 B 样条、4 点立方卷积内插和小波内插分解进行实验仿真并比较。

8.4.1 理想内插

奈奎斯特取样定理说明，按照不低于信号最高频率 2 倍的取样频率对连续信号取样得到的离散信号，可通过理想低通滤波器完全无失真恢复，对应的时域即为无穷多个以取样点值为加权的插值函数 sinc 之和。理想内插核的时域和频域图像如图 8-6 所示。

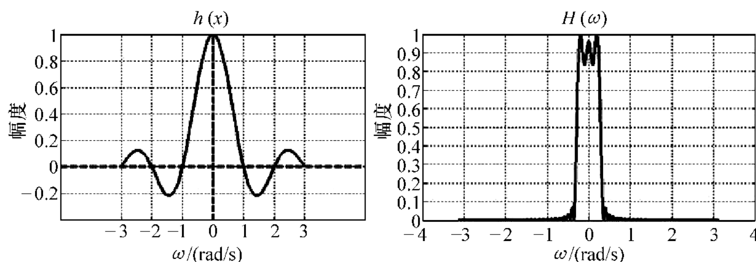


图 8-6 理想内插核的时频和频域图像

内插核 $h(x)$ 只在 $x=0$ 处为 1，其余整数点处为零，从而保证在相同的采样处图像不会被修改。这是理想的内插器，现实中无法真正实现。如图 8-6 所示，其频域由于存在截断而出现波形振荡。取 7 点理想内插器权值，对 sinc 信号进行 1/2 抽取后恢复，原信号及 2∶1 抽取信号如图 8-7 所示。实验结果如图 8-8 所示。

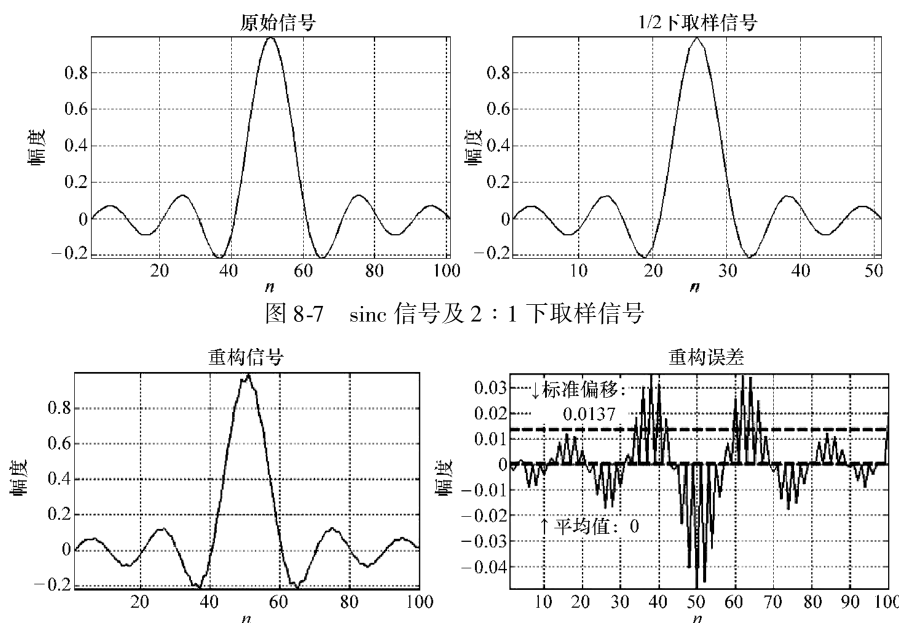


图 8-7 sinc 信号及 2∶1 下取样信号

图 8-8 理想内插实验结果

8.4.2 最近邻域插值

此算法内插的输出像素值由输入图像的最近采样点的值确定，因此又称为点平移算法或像素复制。此算法的内插核函数时域形式定义如下：

$$h(x)=\begin{cases}1 & -0.5<x\leqslant 0.5\\ 0 & \text{其他}\end{cases} \tag{8-78}$$

其频域表达如下：

$$H(\omega)=\mathrm{sinc}\left(\frac{\omega}{2}\right) \tag{8-79}$$

最近邻域插值算法的内插核时域和频域图像如图 8-9 所示。

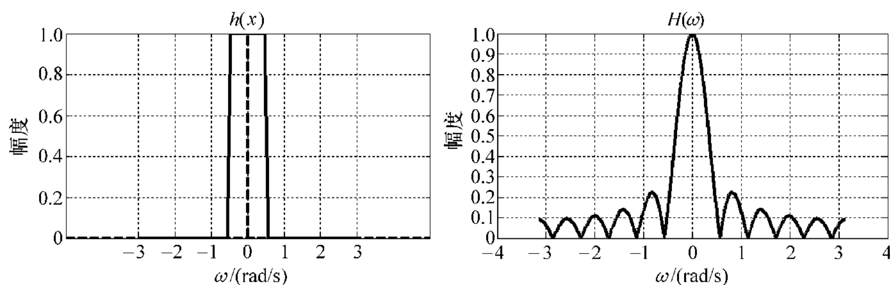


图 8-9　最近邻域内插核的时频图

此算法是一种最简单的逼近 sinc 函数的方法，而 sinc 函数可以做出一个严格的低通滤波器对采样点进行像素的复制和缩放。对 sinc 信号进行内插，实验结果如图 8-10 所示。

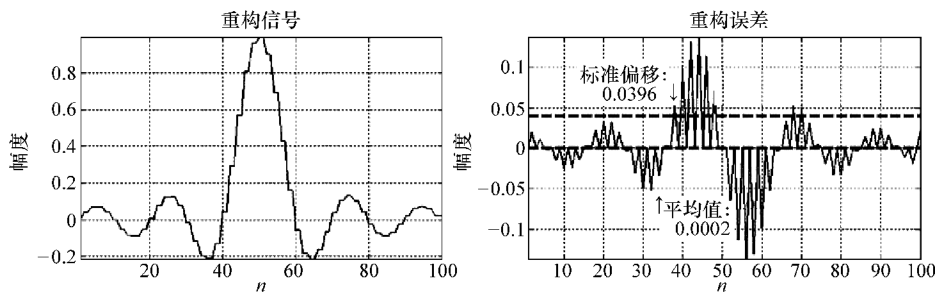


图 8-10　最近邻域内插实验结果

8.4.3 双线性内插

线性内插是一阶内插方法，它对于输入信号每两个间隔点用一条直线连接，内插核幅值的大小只与距离相关。内插核时域表达式如下：

$$h(x)=\begin{cases}1-|x| & 0\leqslant |x|<1\\ 0 & 1\leqslant |x|\end{cases} \tag{8-80}$$

相应的频域表达如下：

$$H(\omega)=\mathrm{sinc}^2\left(\frac{\omega}{2}\right) \tag{8-81}$$

双线性内插核的时频图如图 8-11 所示。这种内插核也叫三角形滤波器或屋顶函数。

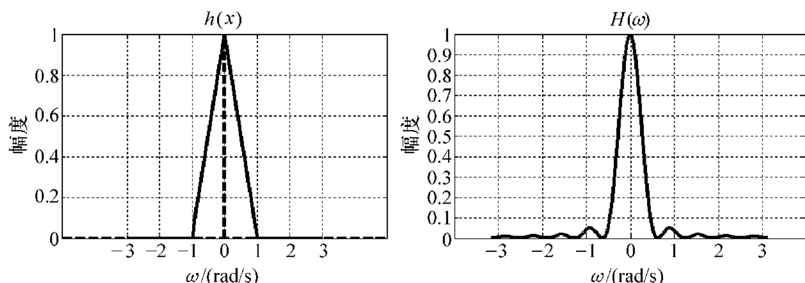


图 8-11　双线性内插核的时频图

由图 8-11 可以看出，线性内插核比最近邻域内插核两边峰值有所下降，阻带内的影响减弱使通带效应加强，从而实现低频滤波。对 sinc 信号进行内插，实验结果如图 8-12 所示。

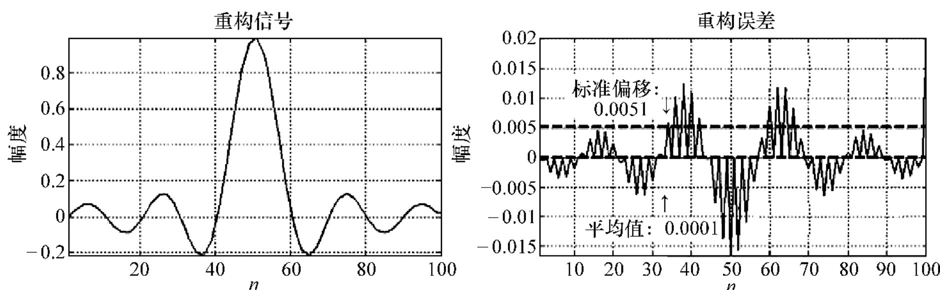


图 8-12　双线性内插实验结果

8.4.4　B 样条内插

B 样条是给定阶数的样条函数加权和，若插值函数 $B(x)$ 是次数不超过 3 的多项式，称之为三次样条插值函数。重构公式如下：

$$\hat{f}(\zeta)=\sum_{k=1}^{K}c_kB_k(\zeta) \tag{8-82}$$

式中，c_k是从输入数据中确定的系数；$B_k(\xi)$ 是样条基函数；K 是给定的数据个数。

n 维 B 样条相当于窄带滤波器 B_0 的 n 重卷积，所以 B_0 是 0 阶 B 样条，B_1 是 1 阶 B 样条，频谱近似为三角滤波器，对应的 B 样条内插相当于线性内插；B_3 则表示 4 个 B_0 的 3 重卷积。B 样条函数构造图如图 8-13 所示：

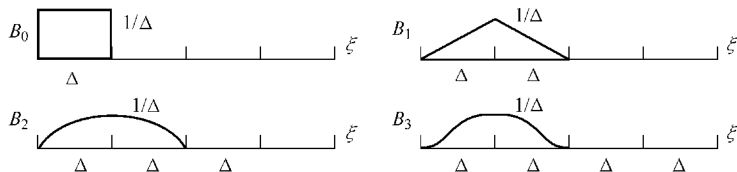


图 8-13　B 样条函数构造示意图

n 阶 B 样条分段多项式如下：

$$B_n(\zeta;\zeta_0,\zeta_1\cdots,\zeta_{n+1}) = (n+1)\sum_{k=0}^{n+1}\frac{(\zeta-\zeta_k)^n U(\zeta-\zeta_k)}{\omega(\zeta_k)} \tag{8-83}$$

式中

$$\omega(\zeta_k) = \prod_{\substack{j=0\\ j\neq k}}^{n+1}(\zeta_k-\zeta_j),\ U(\zeta-\zeta_k) = \begin{cases}1 & \zeta>\zeta_k\\ 0 & \zeta\leqslant\zeta_k\end{cases} \tag{8-84}$$

对于以 1 为步长的等距离结点，插值函数可由下式求出：

$$B_n(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty}\left[\frac{2\sin(u/2)}{u}\right]^n \mathrm{e}^{iux}\mathrm{d}u = \frac{1}{(n-1)!}\delta_n x_+^n \tag{8-85}$$

式中，δ^n 是 n 阶中心差分。$B_n(x)$ 显式表达式如下：

$$(n-1)!\ B_n(x) = \begin{cases}0 & x\leqslant -n/2\\ \left(x+\frac{n}{2}\right)^{n-1} & -n/2\leqslant x\leqslant -n/2+1\end{cases} \tag{8-86}$$

续式（8-86）

$$(n-1)!\,B_n(x)=\begin{cases}\left(x+\frac{n}{2}\right)^{n-1}-\binom{n}{1}\left(x+\frac{n}{2}-1\right)^{n-1} & -\frac{n}{2}+1\leqslant x\leqslant -\frac{n}{2}+2\\ \cdots\cdots & \\ \left(x+\frac{n}{2}\right)^{n-1}-\binom{n}{1}\left(x+\frac{n}{2}-1\right)^{n-1}+\cdots+(-1)^{n-1}\binom{n}{1}\left(x-\frac{n}{2}+1\right)^{n-1} & \frac{n}{2}-1\leqslant x\leqslant\frac{n}{2}\\ 0 & \frac{n}{2}\leqslant x\end{cases}$$

特别地，0 阶 B 样条和 1 阶 B 样条内插核如下：

$$B_1(x) = \begin{cases}0 & x<-1/2\\ 1 & -1/2<x<1/2\\ 0 & 1/2<x\end{cases}\qquad B_2(x) = \begin{cases}0 & x<-1\\ x+1 & -1\leqslant x\leqslant 0\\ -x+1 & 0\leqslant x\leqslant 1\\ 0 & 1\leqslant x\end{cases} \tag{8-87}$$

且 $B_1(\pm 1/2)=1/2$。2 阶 B 样条和 3 阶 B 样条内插核如下：

$$B_3(x)=\begin{cases}0 & x<-3/2\\ \dfrac{1}{2}\left(x+\dfrac{3}{2}\right)^2 & -3/2\leqslant x\leqslant -1/2\\ \dfrac{1}{2}\left(x+\dfrac{3}{2}\right)^2-\dfrac{3}{2}\left(x+\dfrac{1}{2}\right)^2 & -1/2\leqslant x\leqslant 1/2\\ \dfrac{1}{2}\left(-x+\dfrac{3}{2}\right)^2 & 1/2\leqslant x\leqslant 3/2\\ 0 & 3/2\leqslant x\end{cases}$$

$$B_4(x)=\begin{cases}0 & x\leqslant -2,\ x\geqslant 2\\ \dfrac{1}{6}(x+2)^3 & -2\leqslant x\leqslant -1\\ \dfrac{1}{6}(x+2)^3-\dfrac{4}{6}(x+1)^3 & -1\leqslant x\leqslant 0\\ \dfrac{1}{6}(-x+2)^3-\dfrac{4}{6}(-x+1)^3 & 0\leqslant x\leqslant 1\\ \dfrac{1}{6}(-x+2)^3 & 1\leqslant x\leqslant 2\end{cases} \tag{8-88}$$

0 阶 B 样条内插核相当于最近邻域，1 阶 B 样条内插核相当于线性内插，3 阶 B 样条内插核时频图如图 8-14 所示。

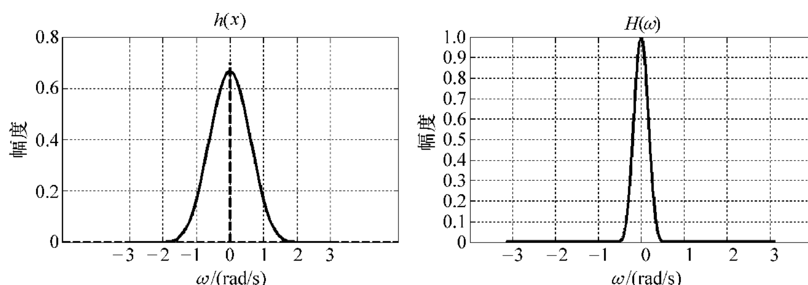


图 8-14 3 阶 B 样条内插器时频图

对 sinc 信号进行重构，实验结果如图 8-15 所示。

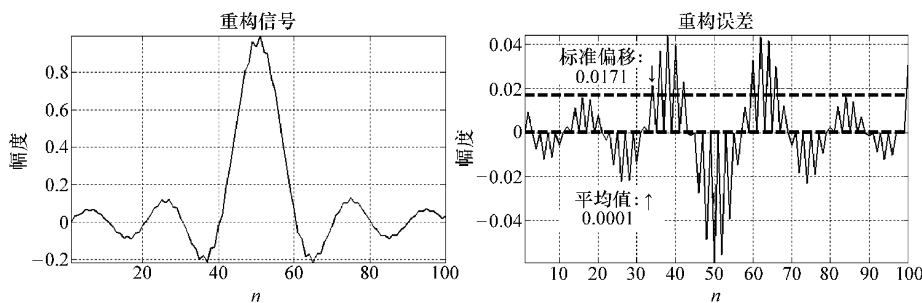


图 8-15 3 阶 B 样条内插实验结果

8.4.5　立方卷积内插

此方法直接利用三次多项式逼近抽样函数 sinc 来构造内插核，常用的是基于两点的三次内插。对称的内插核如下：

$$h_2(x)_{\text{Cubic}}=\begin{cases}A\ |x|^3+B\ |x|^2+C|x|+D & 0\leqslant|x|<1\\ 0 & \text{其他}\end{cases}\tag{8-89}$$

式中，系数 A、B、C、D 可用边界条件来确定。$N=3$ 时的具体内插核如下：

$$h_2(x)_{\text{Cubic}}=2\ |x|^3-3\ |x|^2+1,\ 0\leqslant|x|<1\tag{8-90}$$

立方卷积内插核时频图如图 8-16 所示。

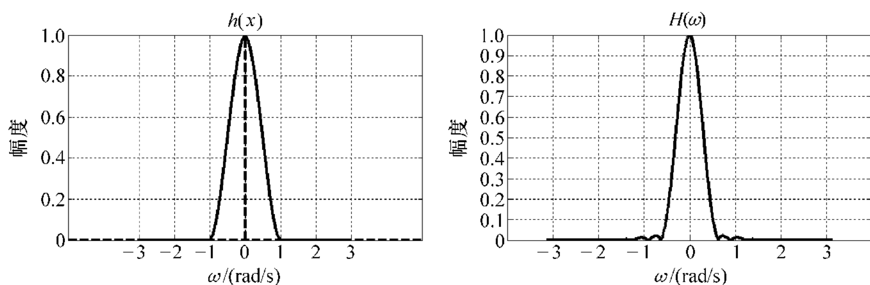


图 8-16　立方卷积内插核时频图（$N=3$）

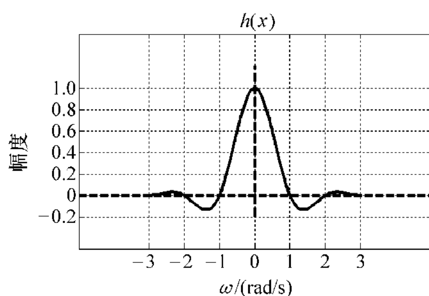
此时立方卷积内插器与线性内插器相同。增加内插数据个数 N 能够在一定程度上提高采样的质量。但需要确定自由度数值，基本原则是判断 $\boldsymbol{h}_{\text{Cubic}}$ 傅里叶变换与理想的矩形滤波器差别，差别越小的内插器内插效果越好。下面给出六点和八点的立方卷积内插器。

$$h_6(x)_{\text{Cubic}}=\begin{cases}6/5\ |x|^3-11/5\ |x|^2+1 & 0\leqslant|x|<1\\ -3/5\ |x|^3+16/5\ |x|^2-27/5|x|+14/5 & 1\leqslant|x|\leqslant2\\ 1/5\ |x|^3-8/5\ |x|^2+21/5|x|-18/5 & 2\leqslant|x|\leqslant3\\ 0 & \text{其他}\end{cases}\tag{8-91}$$

$$h_8(x)_{\text{Cubic}}=\begin{cases}67/56\ |x|^3-123/56\ |x|^2+1 & 0\leqslant|x|<1\\ -33/56\ |x|^3+177/56\ |x|^2-75/14|x|+39/14 & 1\leqslant|x|<2\\ 9/56\ |x|^3-75/56\ |x|^2+51/14|x|-45/14 & 2\leqslant|x|<3\\ -3/56\ |x|^3+33/56\ |x|^2-15/7|x|+18/7 & 3\leqslant|x|<4\\ 0 & \text{其他}\end{cases}\tag{8-92}$$

六点立方内插核的时频图如图 8-17 所示。

对 sinc 信号进行重构，实验结果如图 8-18 所示。



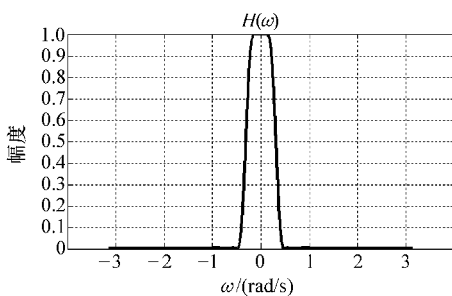
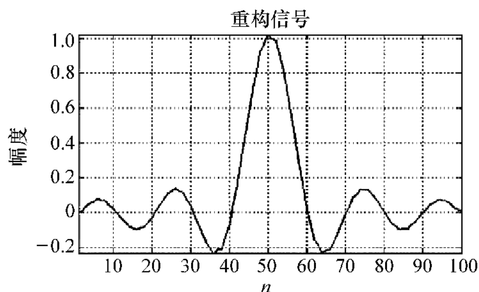


图 8-17 立方卷积内插核时频图（$N=6$）



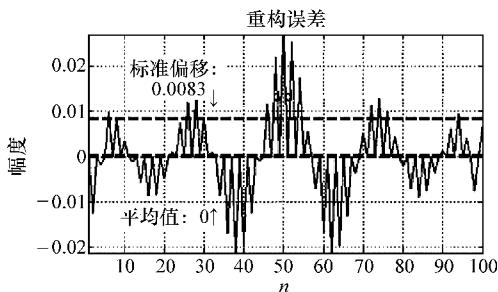


图 8-18 立方卷积内插实验结果（$N=6$）

8.4.6 小波内插

经过一次小波分解，信号可分成低、高频子带系数，其中低频子带的小波系数可以看做是原信号下取样的逼近信号，而高频子带是细节信息。小波插值的主要思想是将待插值的信号作为近似的低频子带系数，根据待插值信号的边缘信息模拟出高频子带系数，通过反变换即可得到图像的 2 的整数次幂的放大。高频系数逼近方法中最简单的就是把高频系数全部置为零，这种方法简单速度快且效果也不错。图 8-19 是小波内插原理图。选用 db2 小波，其低通滤波器系数及滤波器特性如图 8-20 所示。

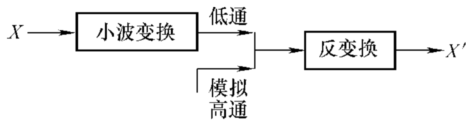
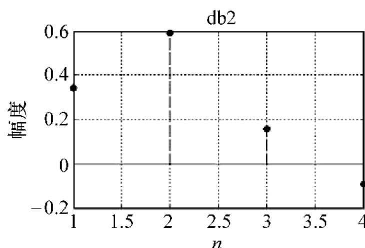


图 8-19 小波内插原理图



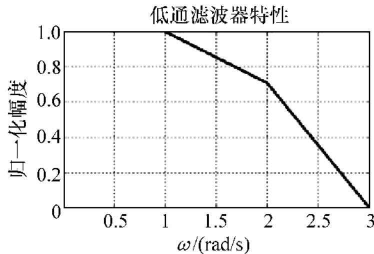


图 8-20 db2 小波低通滤波器

对与上面实验中相同的 Chirp 信号进行插值重建，实验结果如图 8-21 所示。

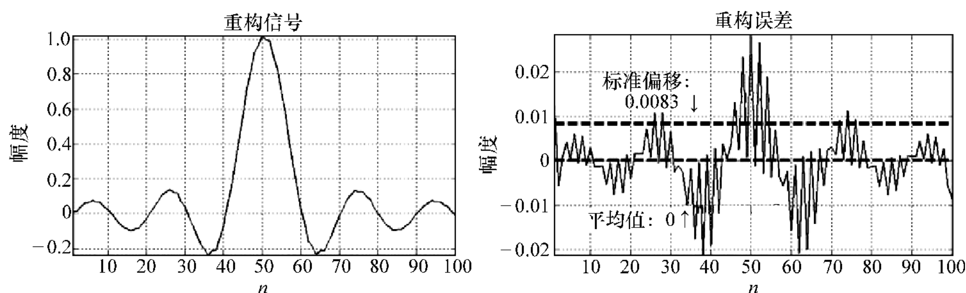


图 8-21　小波内插实验结果

实现上述实验结果的 MATLAB 代码如下：

```
function out = traIP1 (id)

if nargin = =0
    id =1;
end
N =60;d =3;N =60;d =3;sig = sinc(linspace( -5 ,5));

clc,clf,close all
tn ={ 'ideal','nearest','bilinear','B spline','Cubic','Wavelet'};
fn ={ 'idIP(N,d)','neIP(N,d)','biIP(N,d)','bsIP(N,d)','cbIP(N,d)','waIP(N,d)'};

[x w h H] =eval(fn{ id });
toDraw(x,w,d,h,H);
toINP(id,h,sig,x,tn);

if nargout ~ =0
    out = [T];
end

function[x w h H] =idIP(N,d)
x =[ -d:2* d/(N* 1): d];x(x = =0) =eps;h =sin(x* pi)./(x* pi);
nx =length(x);[w H] =toGetH(N,nx,h);

function[x w h H] =neIP(N,d)
x =[ -d:2* d/(N* 1): d];nx =length(x);h =zeros(1,nx);
h((x > -0.5)&(x < =0.5)) =1;[w H] =toGetH(N,nx,h);

function[x w h H] =biIP(N,d)
```

```
x = [ -d:2* d/(N* 1): d];nx = length(x);h = zeros(1,nx);
h((x > -1)&(x < 1)) =1 - abs(x((x > -1)&(x < 1)));
[w H] = toGetH(N,nx,h);

function[x w h H] = bsIP(N,d)
x = [ -d:2* d/(N* 1): d];nx = length(x);h = zeros(1,nx);
sc = (x > = -2)&(x < = -1);
h(sc) = (x(sc) +2).^3/6;
sc = (x > = -1)&(x < =0);
h(sc) = (x(sc) +2).^3/6 -4 * (x(sc) +1).^3/6;
sc = (x > =0)&(x < =1);
h(sc) = ( -x(sc) +2).^3/6 -4 * ( -x(sc) +1).^3/6;
sc = (x > =1)&(x < =2);
h(sc) = ( -x(sc) +2).^3/6;
[w H] = toGetH(N,nx,h);

function[x w h H] = cbIP(N,d)
x = [ -d:2* d/(N* 1): d];nx = length(x);h = zeros(1,nx);
sc = (abs(x) > =0)&(abs(x) < 1);
h(sc) =6 * (abs(x(sc))).^3/5 -11 * (abs(x(sc))).^2/5 +1;
sc = (abs(x) > =1)&(abs(x) < 2);
h(sc) = ( -3)* (abs(x(sc))).^3/5 +16 * (abs(x(sc))).^2/5 - 27 * (abs(x(sc)))/5 +14/5;
sc = (abs(x) > =2)&(abs(x) < 3);
h(sc) = (abs(x(sc))).^3/5 -8 * (abs(x(sc))).^2/5 ...
    + 21 * (abs(x(sc)))/5 -18/5;
[w H] = toGetH(N,nx,h);

function[x w h H] = waIP(N,d)
x = [ -d:2* d/(N* 1): d];w = x;h = x;H = x;
load db2;subplot(1,2,1),set(gca,'FontSize',16);
stem(db2,'k - -','MarkerSize',8,'MarkerFaceColor','k');
grid on;xlabel('\itn');ylabel('amp');title('db2');
subplot(1,2,2),set(gca,'FontSize',16);
plot(abs(fft(db2)),'k','LineWidth',3);grid on;
xlabel('\omega(rad/s)');ylabel('amp');title('Transmission Property of Low
Filter');
set(gca,'XLim',[0 3]);

function[w H] = toGetH(N,nx,h)
w = -pi:2* pi/N^2* pi:pi;nw = length(w);
ws = repmat(w',1,nx);ns = repmat([0:nx -1],nw,1);
```

```
hn = repmat(h,nw,1);en = exp(-j* ws.* ns);H = sum(hn.* en,2);

function toDraw(x,w,d,h,H)
figure,subplot(1,2,1),set(gca,'FontSize',16,'XTick',[-d:d]);
plot(x,h,'k','LineWidth',3),grid on;
set(gca,'XTick',[-d:d],'YTick',[-0.2:0.2:1]);
xr = get(gca,'XLim');xp = [xr(1): 0.01:xr(2)];
hold on,hdl(1)  = plot(xp,zeros(1,length(xp)),'k--');
yr = get(gca,'YLim');yp = [yr(1): 0.01:yr(2)];
hdl(2) = plot(zeros(1,length(yp)),yp,'k--');
set(hdl,'LineWidth',3);
xlabel('\it \omega \rm(rad/s)');ylabel('amp');
title('\ith \rm(\itx \rm)');
subplot(1,2,2),set(gca,'FontSize',16);
plot(w,abs(H)/max(abs(H)),'k','LineWidth',3);grid on;
set(gca,'XTick',[-d:d],'YTick',[0:0.1:1]);
xlabel('\it \omega \rm(rad/s)');ylabel('amp');
title('\itH \rm(\omega)');

function T = toINP(id,h,X,x,tn)

switch id
    case 1
        zps = find(abs(h) < eps);
        nps = [zps(1:end/2) find(h == 1) zps(end/2 +1:end)];
        ips = [(nps(1:end -1) + nps(2:end))/2];Tt = h(ips);
        T = [Tt(1:end/2),1,Tt(end/2 +1:end)];
    case 2
        T = [0 1 1];
    case 3
        T = [h((abs(x) < = 0.5)&(mod(10* x,5) = = 0))];
    case 4
        T = [h((abs(x) < = 1.5)&(mod(10* x,5) = = 0))];
    case 5
        T = [h((abs(x) < = 2.5)&(mod(10* x,5) = = 0))];
    case 6
        disp('Wavelet Interpolation');n = 0;T1 = 1;
    otherwise
        disp('Bad Parameter');
end
T = T1;m = length(T1);
```

```
if m > 3
    T = zeros(1,2* m -3);mid = T1((m +1)/2);
    T(1:2:end) = [T1(1 :(m -1)/2),T1((m +3)/2:end)];
    T(m -1) =mid;
end
n = (length(T) -1)/2;T = T/(sum(T) - T(n +1));

ifnargout = =0
    XX = X;XX(2:2:end) =0;Y = XX;len = length(X);
    Xout = [fliplr(XX(2:n +1)),XX,fliplr(XX(end - n:end -1))];
    for i = n  +2:2:len + n
        Y(i - n) = sum(T.* Xout(i - n:i + n));
    end

    if id = =6
        [cA,cD] = dwt(dyadup(dyaddown(X)),'db2');
        Y =2 * idwt(cA,zeros(size(cA)),'db2');Y = Y(1:length(X));
    end

    figure,subplot(2,2,1),  set(gca,'FontSize',16);
    hdl(1) =plot(X,'k');title('original signal');  axis tight;
    grid on;xlabel('\itn');ylabel('amp');
    subplot(2,2,2),set(gca,'FontSize',16);
    hdl(2) =plot(XX(1:2:end),'k');title('1/2 decimated signal');
    grid on;xlabel('\itn');ylabel('amp');axis tight;
    subplot(2,2,3),  set(gca,'FontSize',16);
    hdl(3) =plot(Y,'k');title(strcat('recovered signal by',tn{id}));
    grid on;xlabel('\itn');ylabel('amp');axis tight;
    subplot(2,2,4),  set(gca,'FontSize',16);
    hdl(4) =plot(Y - X,'k');title('error');
    mv = mean(Y - X);sv = std(Y - X);mvs = ones(1,length(Y))* mv;
    svs = ones(1,length(Y))* sv;hold on;hd(1) =plot(mvs,'k - -');
    hd(2) =plot(svs,'k - -');set(hd,'LineWidth',4);
    grid on;xlabel('\itn');ylabel('amp');axis tight;
    set(hdl,'LineWidth',2);
    mv = fix(mv* 10^4)/10^4;sv = fix(sv* 10^4)/10^4;
    text(length(Y)/3,min(Y - X)/2,strcat('\uparrowmean value:',
    num2str(mv)),'FontSize',16);
    text(length(Y)/3,max(Y - X)/2,strcat('\downarrowstandard deviation
    value:',num2str(sv)),'FontSize',16);
end
```

8.5 全相位插值算法

8.5.1 I 型全相位插值

对于一维有限长的 N 个点进行正交变换可得到 N 个变换域系数，若对此 N 个系数进行反正交变换即可完全恢复原 N 点信号。如果将反变换的时域序号用连续的时间代替，则即可在抽样点上恢复抽样信号，还可以获得其他时间点信号。由于不同的分段起点，可能存在 N 个不同的数据段都包含待插值的样点 $I(n)$，即将产生 N 个不同的插值结果，全相位插值就是取这 N 个插值的均值作为最终结果。

设有限长序列 $x(n)$，取其中任一长度为 N 的子序列 $X(n)$，正交变换 $\boldsymbol{\alpha}$ 和反正正交变换 $\boldsymbol{\beta}$ 过程如下：

$$A(l) = \sum_{n=0}^{N-1}\alpha(l,n)X(n),\ X(m) = \sum_{l=0}^{N-1}\beta(m,l)A(l) \tag{8-93}$$

则连续的插值信号可由下式产生：

$$\begin{aligned}\hat{X}(t) &= \sum_{l=0}^{N-1}\beta(t,l)A(l) = \sum_{l=0}^{N-1}\beta(t,l)\sum_{n=0}^{N-1}\alpha(l,n)X(n)\\ &= \sum_{n=0}^{N-1}\left(\sum_{l=0}^{N-1}\beta(t,l)\alpha(l,n)\right)X(n)\\ &= \sum_{n=0}^{N-1}H(t,n)X(n)\end{aligned} \tag{8-94}$$

式中
$$H(t,n) = \sum_{l=0}^{N-1}\beta(t,l)\alpha(l,n)$$

Walsh 变换基只有 ±1 离散值，傅里叶变换基有复数，因此选择 DCT 基作为插值的正交变换。正反变换定义如下：

$$\beta(t,l) = \begin{cases}\sqrt{\dfrac{1}{N}} & l=0,\ 0\leqslant t\leqslant N-1\\ \sqrt{\dfrac{2}{N}}\cos\dfrac{(2t+1)l\pi}{2N} & 1\leqslant l\leqslant N-1,\ 0\leqslant t\leqslant N-1\end{cases} \tag{8-95}$$

$$\alpha(l,n) = \begin{cases}\sqrt{\dfrac{1}{N}} & l=0,\ 0\leqslant n\leqslant N-1\\ \sqrt{\dfrac{2}{N}}\cos\dfrac{(2n+1)l\pi}{2N} & 1\leqslant l\leqslant N-1,\ 0\leqslant n\leqslant N-1\end{cases} \tag{8-96}$$

对于序列 $\boldsymbol{x}$ 中包含 $x(n)$ 和 $x(n+1)$ 的任意内插点 $x(n+\tau)$ 的分段共有 $N-1$

个，其中 $0 \leqslant \tau < 1$，如图 8-22 所示。

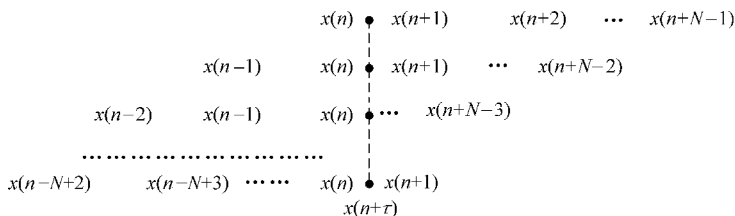


图 8-22　包含内插点的 $N-1$ 个数据段

每段生成的插值点结果为

$$
\begin{aligned}
\hat{X}_0(\tau) &= \sum_{j=0}^{N-1} H(\tau, j) X_0(j) = \sum_{j=0}^{N-1} H(\tau, j) x(n+j) \\
\hat{X}_1(\tau) &= \sum_{j=0}^{N-1} H(1+\tau, j) X_0(j) = \sum_{j=0}^{N-1} H(1+\tau, j) x(n-1+j) \\
&\vdots \\
\hat{X}_{N-2}(\tau) &= \sum_{j=0}^{N-1} H(N-2+\tau, j) X_{N-2}(j) = \sum_{j=0}^{N-1} H(N-2+\tau, j) x(n-N+2+j)
\end{aligned}
\tag{8-97}
$$

全相位内插是所有分段产生的内插结果和的平均值，即全相位内插值等于

$$
X_n(\tau) = \frac{1}{N-1} \sum_{i=0}^{N-2} \hat{X}_i(\tau) = \frac{1}{N-1} \sum_{i=0}^{N-2} \sum_{j=0}^{N-1} H(i+\tau, j) x(n-i+j) \tag{8-98}
$$

令 $k=i-j$ 并交换求和顺序可得到：

$$
\begin{aligned}
X_n(\tau) &= \frac{1}{N-1} \sum_{k=0}^{N-2} x(n-k) \sum_{i=k}^{N-2} H(i+\tau, i-k) + \\
&\quad \frac{1}{N-1} \sum_{k=-N+1}^{-1} x(n-k) \sum_{i=0}^{N-1+k} H(i+\tau, i-k) \\
&= \sum_{-N+1}^{N-2} x(n-k) h(\tau, k)
\end{aligned}
\tag{8-99}
$$

式中

$$
h(\tau, k) = \begin{cases} \dfrac{1}{N-1} \displaystyle\sum_{i=k}^{N-2} H(i+\tau, i-k) & k = 0, 1, 2\cdots, N-2 \\ \dfrac{1}{N-1} \displaystyle\sum_{i=0}^{N-1+k} H(i+\tau, i-k) & k = -N+1, -N+2, \cdots, -1 \end{cases} \tag{8-100}
$$

按照式（8-99）写出 $N=4$ 时的六点内插公式如下：

$$
X_n(\tau) = \sum_{-3}^{2} x(n-k) h(\tau, k)
$$

由式（8-100）可写出内插核分段函数如下：

$$h(\tau, k)=\begin{cases} h(\tau, 2)=\frac{1}{3}H(2+\tau, 0) \\ h(\tau, 1)=\frac{1}{3}H(1+\tau, 0)+\frac{1}{3}H(2+\tau, 1) \\ h(\tau, 0)=\frac{1}{3}H(\tau, 0)+\frac{1}{3}H(1+\tau, 1)+\frac{1}{3}H(2+\tau, 2) \\ h(\tau, -1)=\frac{1}{3}H(\tau, 1)+\frac{1}{3}H(1+\tau, 2)+\frac{1}{3}H(2+\tau, 3) \\ h(\tau, -2)=\frac{1}{3}H(\tau, 2)+\frac{1}{3}H(1+\tau, 3) \\ h(\tau, -3)=\frac{1}{3}H(\tau, 3) \end{cases} \tag{8-101}$$

以 x 表示各数据段相对于插值点的距离，即 $x=k-\tau$，重写式（8-101）如下：

$$h_6(x)=\begin{cases} \frac{1}{3}H(4-x, 0) & 1\leqslant x<2 \\ \frac{1}{3}H(2-x, 0)+\frac{1}{3}H(3-x, 1) & 0\leqslant x<1 \\ \frac{1}{3}H(-x, 0)+\frac{1}{3}H(1-x, 1)+\frac{1}{3}H(2-x, 2) & -1<x<0 \\ \frac{1}{3}H(-1-x, 1)+\frac{1}{3}H(-x, 2)+\frac{1}{3}H(1-x, 3) & -2<x\leqslant -1 \\ \frac{1}{3}H(-2-x, 2)+\frac{1}{3}H(-1-x, 3) & -3<x\leqslant -2 \\ \frac{1}{3}H(-3-x, 3) & -4<x\leqslant -3 \end{cases} \tag{8-102}$$

同理可写出 $N=5$ 时的八点内插核分段函数为

$$h_8(x)=\begin{cases} \frac{1}{4}H(4-x, 4) & 3\leqslant x<4 \\ \frac{1}{4}[H(3-x, 3)+H(4-x, 4)] & 2\leqslant x<3 \\ \frac{1}{4}[H(2-x, 2)+H(3-x, 3)+H(4-x, 4)] & 1\leqslant x<2 \\ \frac{1}{4}[H(1-x, 1)+H(2-x, 2)+H(3-x, 3)+H(4-x, 4)] & 0\leqslant x<1 \\ \frac{1}{4}[H(-x, 0)+H(1-x, 1)+H(2-x, 2)+H(3-x, 3)] & -1\leqslant x<0 \\ \frac{1}{4}[H(-x, 0)+H(1-x, 1)+H(2-x, 2)] & -2\leqslant x<-1 \\ \frac{1}{4}[H(-x, 0)+H(1-x, 1)] & -3\leqslant x<-2 \\ \frac{1}{4}H(-x, 0) & -4\leqslant x<-3 \end{cases} \tag{8-103}$$

八点全相位内插核时频图如图 8-23 所示。

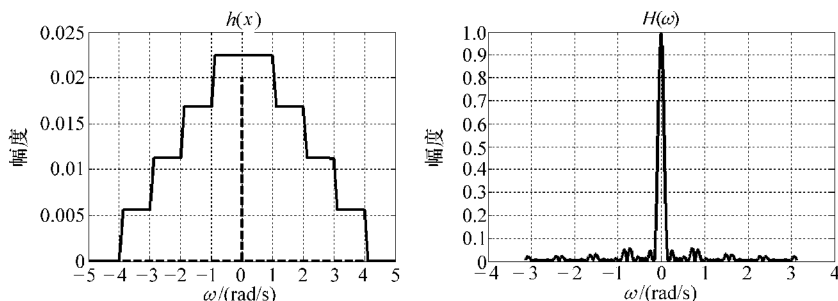


图 8-23　八点全相位 DCT 内插核时频图（$N=5$）

对与上面实验中相同的 Chirp 信号进行插值重建，实验结果如图 8-24 所示。

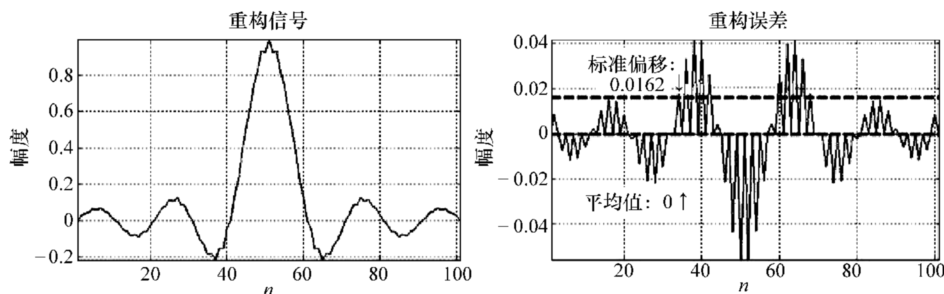


图 8-24　全相位 I 型内插实验结果

实现上述结果的 MATLAB 代码［只列出 $h(x)$ 生成函数］如下：

```
function[t w h H] =ap1IP(N,d)
t =[ -d +2* d/N:2* d/N:d];L =[0: d -1];N =length(t)* d;
beta =sqrt(2/N)* cos((2* t +1)'* L* pi/(2* N));
beta(:,1) =1/sqrt(N);
alpha =sqrt(2/N)* cos((2* L +1)'* L* pi/(2* N));
alpha(:,1) =1/sqrt(N);
H =beta* alpha';h =zeros(size(t));
rg0 = (t > 0)&(t <=1);rg1 = (t > 1)&(t <=2);
rg2 = (t > 2)&(t <=3);rg3 = (t > 3)&(t <=4);
rg_1 = (t > -1)&(t <=0);rg_2 = (t > -2)&(t <= -1);
rg_3 = (t > -3)&(t <= -2);rg_4 = (t > -4)&(t <= -3);
Hrg0 = (t >=0)&(t < 1);Hrg1 = (t >=1)&(t < 2);
Hrg2 = (t >=2)&(t < 3);Hrg3 = (t >=3)&(t < 4);

h(rg3) =1/4* H(Hrg0,5);
```

```
h(rg2)=1/4 * (H(Hrg0,4)+H(Hrg1,5));
h(rg1)=1/4 * (H(Hrg0,3)+H(Hrg1,4)+H(Hrg2,5));
h(rg0)=1/4 * (H(Hrg0,2)+H(Hrg1,3)+H(Hrg2,4)+H(Hrg3,5));
h(rg_1)=1/4 * (H(Hrg0,1)+H(Hrg1,2)+H(Hrg2,3)+H(Hrg3,4));
h(rg_2)=1/4 * (H(Hrg1,1)+H(Hrg2,2)+H(Hrg3,3));
h(rg_3)=1/4 * (H(Hrg2,1)+H(Hrg3,2));
h(rg_4)=1/4* H(Hrg3,1);
```

8.5.2　Ⅱ型全相位插值

Ⅰ型全相位插值建立在正交基的扩展尺寸上，即把内插点看作正交基新增变换点的函数。为了保证数据的相位不变，这种方法限定了正交基的类型，DFT 和 DWT 就无法得到正确的插值形式。Ⅱ型全相位克服了这个问题，其基本思想是建立全相位低通滤波器，并按照二维全相位模板形成的方法求出对应的一维模板。全相位低通滤波器的设计在第 2 章中已经讨论过，正交基选择度很大，而且可以通过加窗进一步改善内插性能。设计步骤如下：

1）构造传输特性序列 $\boldsymbol{H}$；

2）选择无窗、单窗和双窗类型的基窗；

3）按照式（8-70）、式（8-75）构造 DWT、DFT 和 DCT 模板 T。

以 $N=5$ 时设计 DFT 模板为例，设定低通序列 $\boldsymbol{H}=[1\ 0.5\ 0\ 0\ 0.5]$，单窗和双窗类型选择三角窗作为基窗，设计得到的 3 种全相位模板分别为

$\boldsymbol{T}_{\mathrm{n}}=[0.0873\quad 0.0255\quad 0.0382\quad 0.3491\quad 0.6667\quad 0.3491\quad 0.0382\quad 0.0255\quad 0.0873]$

$\boldsymbol{T}_{\mathrm{s}}=[0.0485\quad 0.0212\quad 0.0424\quad 0.3879\quad 0.6667\quad 0.3879\quad 0.0424\quad 0.0212\quad 0.0485]$

$\boldsymbol{T}_{\mathrm{d}}=[0.0263\quad 0.0153\quad 0.0383\quad 0.4201\quad 0.7622\quad 0.4201\quad 0.0383\quad 0.0153\quad 0.0263]$

DFT 基三种模板的时频图如图 8-25 所示。

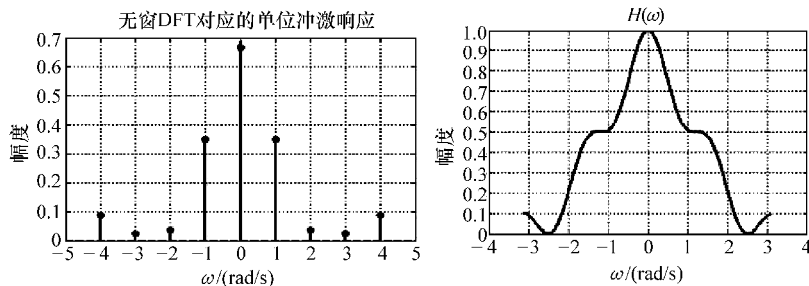


图 8-25　九点Ⅱ型无窗、单窗和双窗全相位 DFT 内插器时频图（$N=5$）

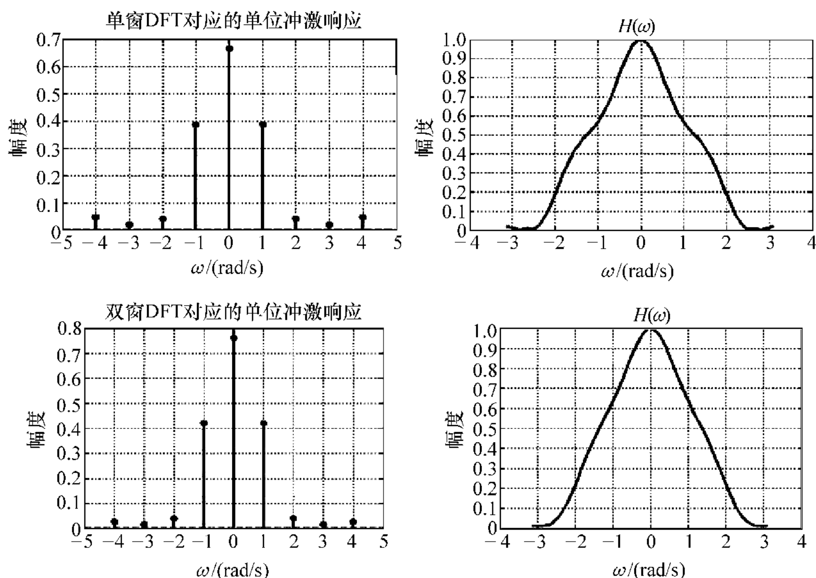


图 8-25　九点Ⅱ型无窗、单窗和双窗全相位 DFT 内插器时频图（$N=5$）(续)

按照同样方法可求得 DWT 三种模板的时频图如图 8-26 所示。三种模板为

$\boldsymbol{T}_n=[0.0614\quad 0\quad 0.0337\quad 0.4049\quad 0.8435\quad 0.4049\quad 0.0337\quad 0\quad 0.0614]$；

$\boldsymbol{T}_s=[0.0735\quad 0.0224\quad -0.0449\quad 0.4490\quad 1.0102\quad 0.4490\quad -0.0449\quad 0.0224\quad 0.0735]$；

$\boldsymbol{T}_d=[0.0706\quad 0\quad 0.0204\quad 0.4089\quad 0.9711\quad 0.4089\quad 0.0204\quad 0\quad 0.0706]$。

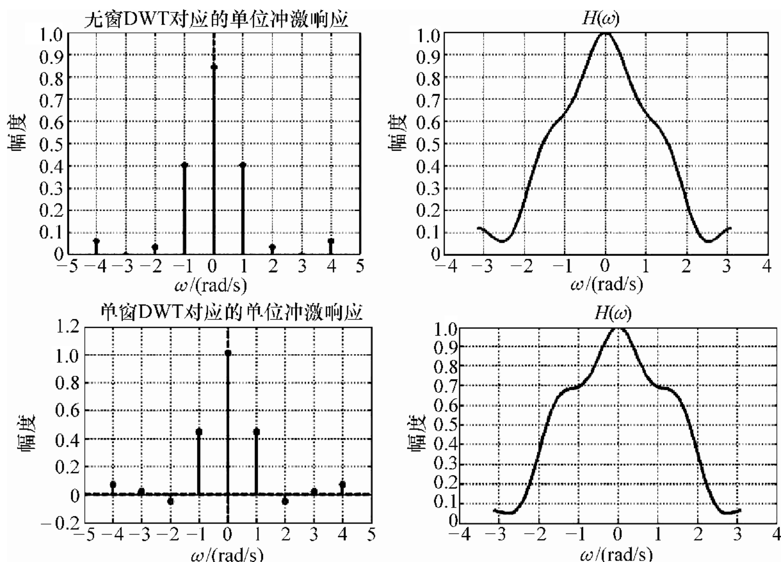


图 8-26　九点Ⅱ型无窗、单窗和双窗全相位 DWT 内插器时频图（$N=5$）

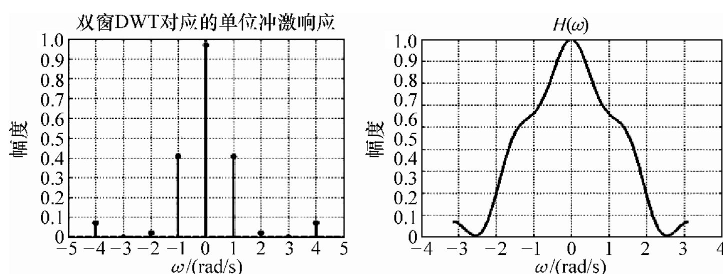


图 8-26　九点Ⅱ型无窗、单窗和双窗全相位 DWT 内插器时频图（$N=5$）(续)

DCT3 种模板的时频图如图 8-27 所示。三种模板为

$\boldsymbol{T}_n=[-0.0229\quad -0.0601\quad 0.0642\quad 0.5188\quad 0.5103\quad 0.5188\quad 0.0642\quad -0.0601\quad -0.0229]$；

$\boldsymbol{T}_s=[-0.0209\quad -0.0456\quad 0.0649\quad 0.5015\quad 0.4143\quad 0.5015\quad 0.0649\quad -0.0456\quad -0.0209]$；

$\boldsymbol{T}_d=[-0.0228\quad -0.0315\quad 0.0590\quad 0.4952\quad 0.4189\quad 0.4952\quad 0.0590\quad -0.0315\quad -0.0228]$。

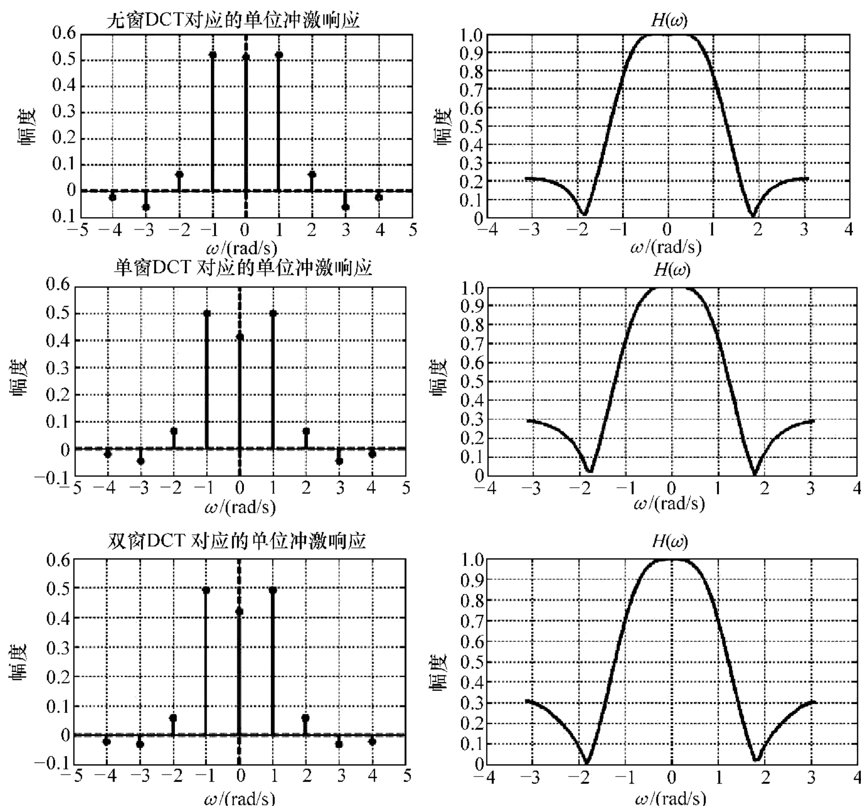


图 8-27　九点Ⅱ型无窗、单窗和双窗全相位 DCT 内插器时频图（$N=5$）

一般 N 阶的Ⅱ型全相位内插核的点数为 $2N-1$ 个且具有严格的线性相位。内插时只需将中心点对准待插位置，模板其余各点依次与相邻已知信号点相乘并求和即得内插结果。分别使用上面 9 种模板对 sinc 信号进行 1/2 抽取恢复，实验结果如图 8-28 所示。

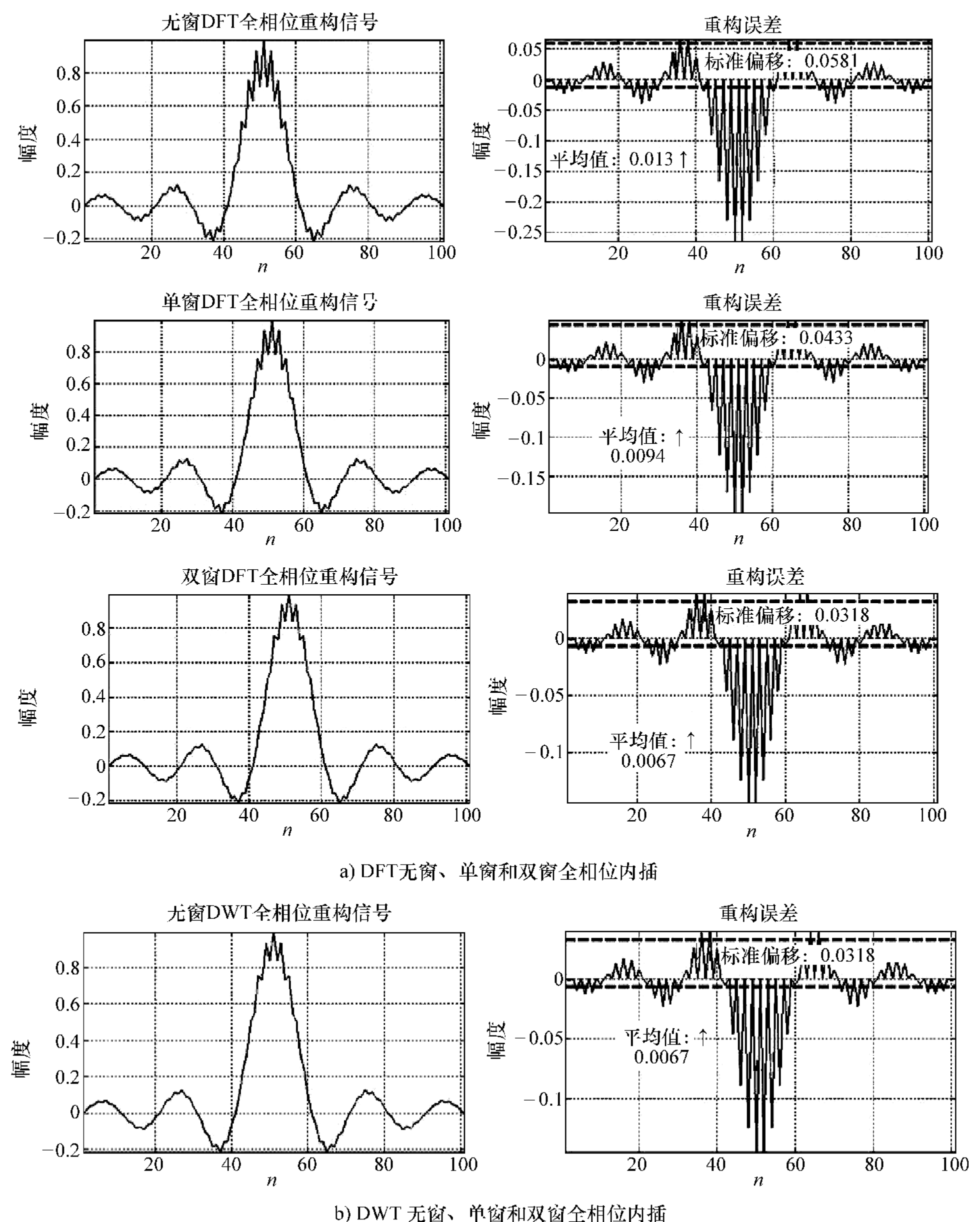
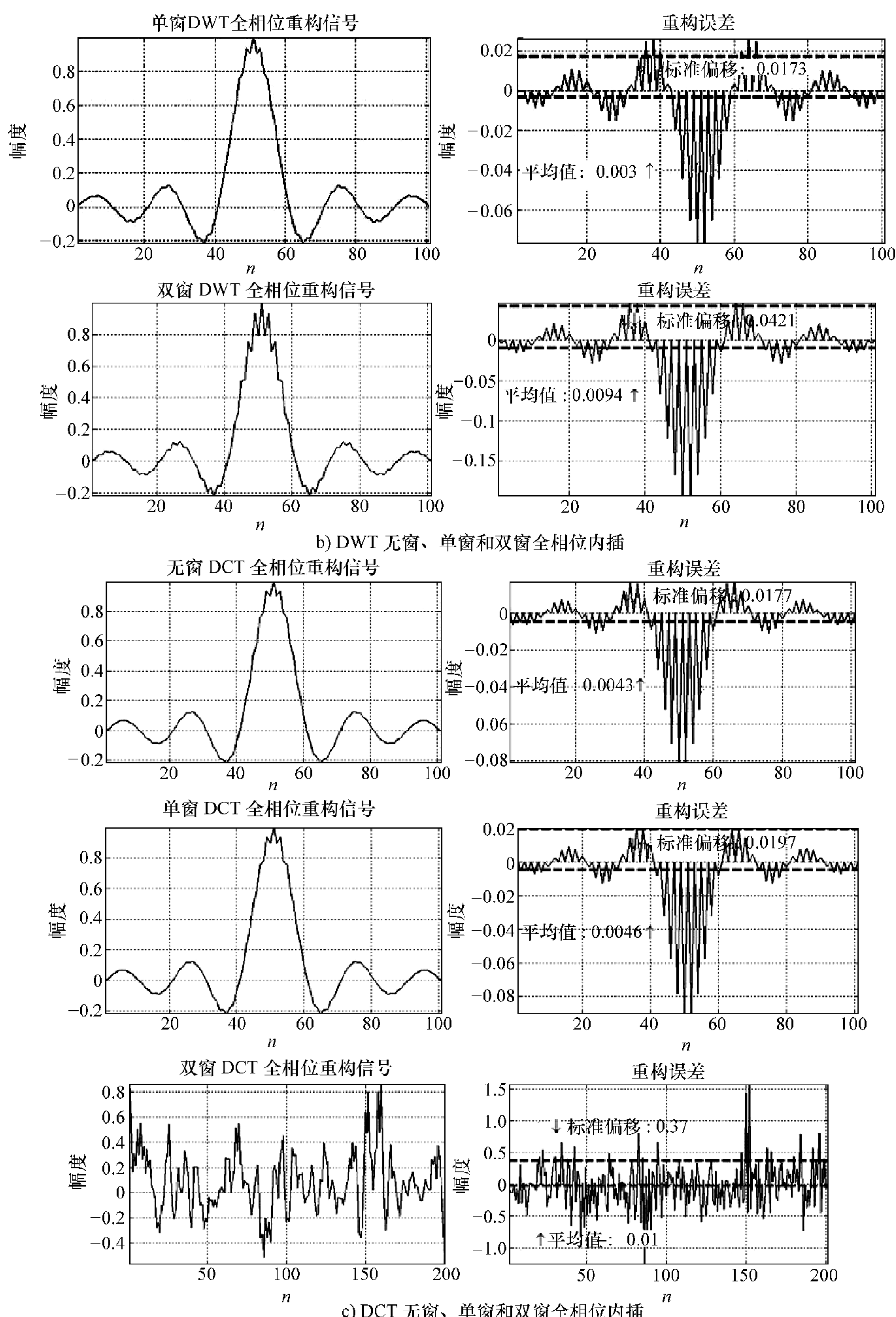


图 8-28　九点Ⅱ型无窗、单窗和双窗全相位内插实验结果



c) DCT 无窗、单窗和双窗全相位内插

图 8-28 九点Ⅱ型无窗、单窗和双窗全相位内插实验结果（续）

实现上述结果的 MATLAB 代码如下：

```
function[t w h H tl] =ap2IP(N,d)
fn ={ 'ap2DFT(N,tp)','ap2DWT(N,tp)','ap2DCT(N,tp)'};
bn ={ 'of DFT','of DWT','of DCT'};wn ={ 'none -win','single -win','dual -win'};
for id =3:3
    for tp =3:3
        [t w h H] =eval(fn{ id });
    end
end
tl =strcat(bn{ id },wn{ tp });

function[t w h H] =ap2DFT(N,tp)
PB =fix(N/4);H =zeros(1,N);H(1:PB) =1;H(PB +1) =0.5;
H(end -PB +1:end) =H(PB +1: -1:2);h0 =ifft(H,N);
switch tp
    case 1
        win =conv(boxcar(N),boxcar(N));
    case 2
        win =conv(triang(N),boxcar(N));
    case 3
        win =conv(triang(N),triang(N));
    otherwise
        error('bad parameters');
end
win =win/max(win);h =[h0(2:end),h0].* win';t =[ -N +1 :N -1];
h =h/(sum(h) -h(N));[w H] =toGetH(N,h);

function[t w h H] =ap2DWT(N,tp)
path(path,'D:\work\book1\chapter 4');
T =apDWT(N);h =T(:,tp)';h =[h(end: -1:2),h];
h =h/(sum(h) -h(N));  t =[ -N +1:N -1];
[w H] =toGetH(N,h);

function[t w h H] =ap2DCT(N,tp)
path(path,'D:\work\book1\chapter 4');
T =apDCT(N);h =T(:,tp)';h =[h(end: -1:2),h];
h =h/(sum(h) -h(N));t =[ -N +1:N -1];
[w H] =toGetH(N,h);

function[w H] =toGetH(N,h)
w = -pi:2* pi/16^2* pi:pi;nw =length(w);  nx =length(h);
```

```
ws = repmat(w',1,nx);ns = repmat([0:nx -1],nw,1);
hn = repmat(h,nw,1);en = exp(-1i* ws.* ns);H = sum(hn.* en,2);
```

以上针对 2：1 下取样信号采用全相位内插实现了信号重建。对于更高比例的下取样信号，Ⅰ型 DCT 全相位内插器可以通过增加 $\boldsymbol{\beta}$ 矩阵的行数来求取模板，对于Ⅱ型全相位内插器有两种方式来实现：一是设计更大长度的传输特性 $\boldsymbol{H}$，这样内插需要参考的信号也相应地增加，但与内插点距离增大的像素之间的相关性也降低，因此这种方式的内插效果并不十分理想；二是将内插点看作窗的函数，通过窗函数映射不同位置内插点的加权值而构造模板。

8.6　图像内插实验

利用上面介绍的 16 种内插器即理想、最近邻域、双线性、三次 B 样条、立方卷积、小波、全相位Ⅰ型，全相位Ⅱ型（共 9 种）对 3 幅 2：1 下取样图像进行重构实验，其中全相位Ⅱ型除了行列方向使用一维模板分别进行内插恢复外，又可以设计二维模板直接内插重构，因此对应的内插方式共有 27 种。实验路线图如图 8-29 所示。

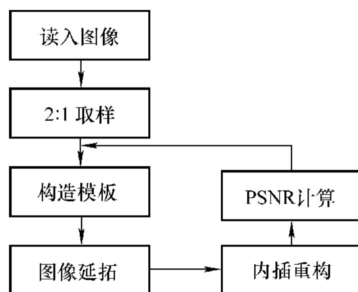


图 8-29　内插实验流程

测试使用的 3 种图像及谱图如图 8-30 所示。

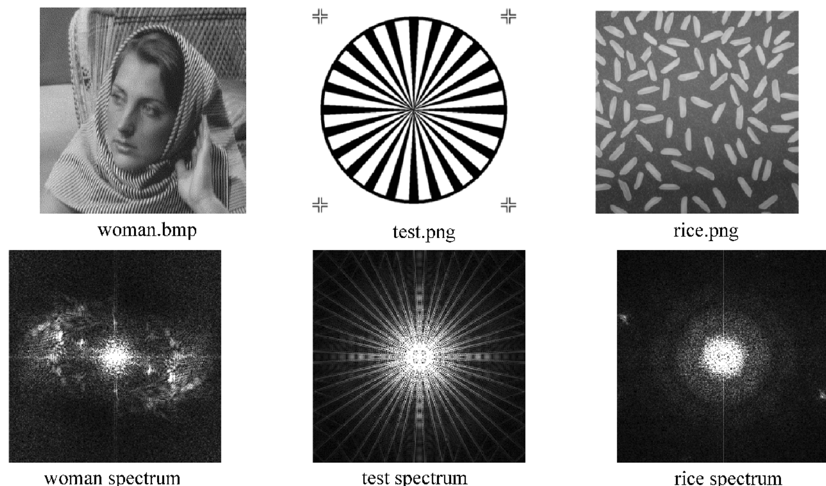


图 8-30　测试图像及谱分布

以 woman. bmp 为例，画出各种内插结果图（见图 8-31），其中全相位Ⅱ型只画出了双窗结果。

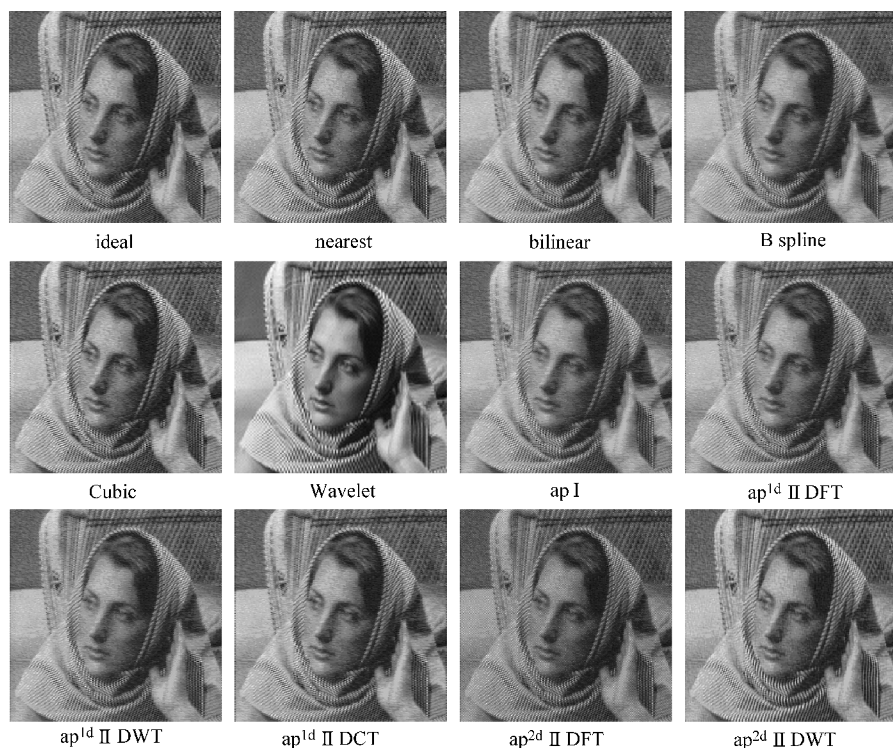


图 8-31　12 种内插器重构图像

设计得到的 3 种二维加窗模板（只列出双窗时结果）如下：

$$
\boldsymbol{T}_{\mathrm{DFT}}^{2\mathrm{d}}=\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0.0714 & 0.0714 & 0.0714 & 0 & 0\\
0 & 0 & 0.2857 & 1 & 0.2857 & 0 & 0\\
0 & 0 & 0.0714 & 0.0714 & 0.0714 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

$$
\boldsymbol{T}_{\mathrm{DWT}}^{2\mathrm{d}}=\begin{bmatrix}
0 & 0 & 0 & 0.0385 & 0 & 0 & 0\\
0 & 0 & -0.0577 & 0 & -0.0577 & 0 & 0\\
0 & -0.0577 & 0 & 0.3269 & 0 & -0.0577 & 0\\
0.0385 & 0 & 0.3269 & 1.0769 & 0.3269 & 0 & 0.0385\\
0 & -0.0577 & & 0.3269 & & -0.0577 & 0\\
0 & 0 & -0.0577 & 0 & -0.0577 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

$$
T_{\mathrm{DCT}}^{2\mathrm{d}}=\begin{bmatrix}
0.0021 & 0.0042 & -0.0064 & 0.0099 & -0.0064 & 0.0042 & 0.0021\\
0.0042 & -0.0127 & -0.0466 & 0.0383 & -0.0466 & -0.0127 & 0.0042\\
-0.0064 & -0.0466 & 0.0127 & 0.2972 & 0.0127 & -0.0466 & -0.0064\\
0.0099 & 0.0383 & 0.2972 & 0.8310 & 0.2972 & 0.0383 & 0.0099\\
-0.0064 & -0.0466 & 0.0127 & 0.2972 & 0.0127 & -0.0466 & -0.0064\\
0.0042 & -0.0127 & -0.0466 & 0.0383 & -0.0466 & -0.0127 & 0.0042\\
0.0021 & 0.0042 & -0.0064 & 0.0099 & -0.0064 & 0.0042 & 0.0021
\end{bmatrix}
$$

上面三种模板的时频图如图 8-32 所示。

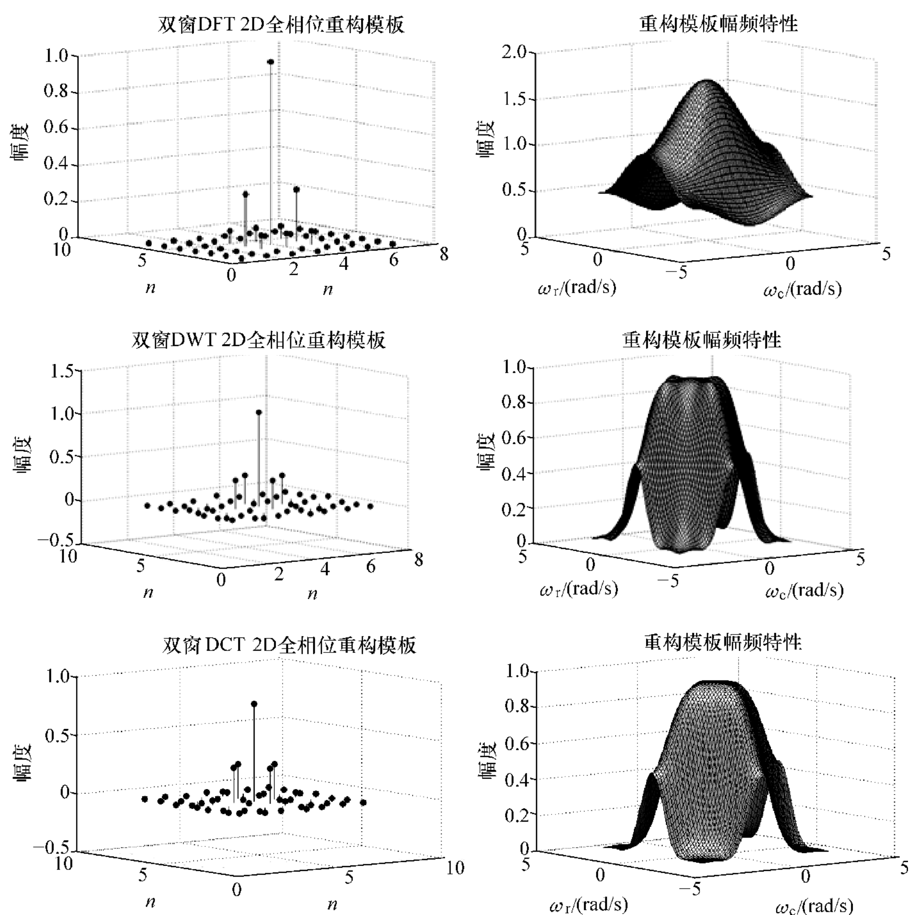


图 8-32　ap^{2d}内插核时频图

各种算法重构得到的 PSNR 对比结果如表 8-1，表 8-2 所示。

表 8-1 传统方法内插

Method / images	Ideal	Nearest	Bi-linear	B-Spline	Cubic	Wavelet
Woman	20. 30	19. 94	21. 63	22. 44	21. 18	30. 30
test	21. 45	17. 45	22. 07	19. 90	21. 41	33. 72
rice	28. 86	26. 16	29. 45	28. 30	28. 77	34. 44

表 8-2 全相位方法内插

Catalogue	Ⅱ ap1									Ⅱ ap2								
base	DFT			DWT			DCT			DFT			DWT			DCT		
window	N	S	D	N	S	D	N	S	D	N	S	D	N	S	D	N	S	D
woman	22. 23	22. 18	22. 07	22. 14	21. 67	22. 08	21. 73	21. 82	21. 82	15. 68	15. 65	18. 31	25. 53	25. 52	25. 49	25. 37	24. 40	23. 59
test	18. 65	19. 93	20. 78	20. 21	20. 39	20. 12	21. 71	21. 78	21. 82	11. 41	11. 65	14. 81	25. 03	25. 33	25. 41	24. 91	22. 90	21. 48
rice	27. 11	28. 14	28. 76	28. 31	28. 33	28. 22	29. 13	29. 24	29. 30	16. 73	16. 81	20. 11	31. 90	32. 03	32. 13	31. 67	28. 85	27. 14

对比实验结果可以得到以下结论，3 种图像中具有丰富低频成分的 rice. png 重构效果最好，而高频成分较多的 test. png 效果最差；采用 DWT 和 DCT 的二维全相位内插效果基本高出传统方法（小波除外）至少 3dB；在一维方面，效果基本与双线性内插和三次 B 样条效果相当。需要指出的是，全相位模板内插效果与窗函数的选择有关，上面实验中全部采用三角窗，如果根据图像谱分布情况来自适应选择窗函数以达到优化内插效果的目的，性能将进一步得到提升。

实现上述实验结果的 MATLAB 代码如下：

```
function out = apIP2(k,m)

ifnargin = =0
   k =3;m =9;clc,clf,close all
end
imgf ={ 'woman.bmp','test.png','rice.png'};
path(path,'D:\work\book1\chapter 7\images');
path(path,'D:\work\book1\chapter 5');

tn ={ 'ideal','nearest','bilinear','B spline','Cubic','Wavelet','ap1','ap2',
'ap3'};
[X,R,C] =toPreProcess(imgf{ k });
for n =m:m
   switch n
      case { 1 2 3 4 5 }
         T =traIP1(n);
      case 6
```

```
            break;
        case 7
            T = apIP1(1, -1, -1);
        case 8
            for id = 3:3
                for wp = 3:3
                    T = apIP1(2,id,wp);
                end
            end
        case 9
            for id = 3:3
                for wp = 3:3
                    T = apIP3(id,wp);
                end
            end
        otherwise
            disp('Invalid Type');
    end
    toINP(T,X,R,C);
end

if n = =6
    [cA1,cH1,cV1,cD1] = dwt2(X,'db4');
    cas = size(cA1);
    A0 = idwt2(cA1,zeros(cas),zeros(cas),zeros(cas),'db4',[R,C]);
    % figure ,imshow(X,[]);figure,imshow(A0,[]);
    ERR = sum(sum((uint8(A0) - X).^2))/numel(X);
    disp('PSNR');10* log10(255^2/ERR)
end

ifnargout ~ =0
    out = [h;H];
end

function T = apIP3(id,wp)
bn = { 'of DFT','of DWT','of DCT'};N = 4;mid = N;
wn = { 'ap2d none - win','ap2d single - win','ap2d dual - win'};
switch id
    case 1
        H = [1 1 0.5 1;1 0.5 0 0.5;0.5 0 0 0;1 0.5 0 0.5];
        Tout = ap2nsdDFT(N,H);T = Tout{ 1 }(:,:,wp);
```

```
    case 2
        H=[1 1 1 0.5;1 1 0.5 0;1 0.5 0 0;0.5 0 0 0];
        Tout=ap2nsdDWT(N,H);T=Tout{1}(:,:,wp);
    case 3
        H=[1 1 1 0.5;1 1 0.5 0;1 0.5 0 0;0.5 0 0 0];
        Tout=ap2nsdDCT(N,H);T=Tout{1}(:,:,wp);
    otherwise
        disp('Bad Parameter');
end
T=T/(sum(sum(T))-T(mid,mid));
subplot(1,2,1),stem3(T,'k','fill');view(-37.5,10);
set(gca,'FontSize',16);box off;
xlabel('\itn');ylabel('\itn');zlabel('amplitude');
title(strcat(strcat('\it\bfT\rm',bn{id}),wn{wp}));
wr=-pi:pi/2^5:pi;wc=-pi:pi/2^5:pi;
subplot(1,2,2),h1=surf(Tout{2}(:,:,wp));
colormap([1 1 1]);light('Position',[-4 -4 1],'Style','Infinite');lighting
gouraud;
set(h1, 'FaceLighting','phong','EdgeColor','k','Facecolor','interp', 'XDa-
ta',wc,'YData',wr);
set(gca,'FontSize',16);view(-37.5,10);
xlabel('\it\omega\rm_c(rad/s)');ylabel('\it\omega\rm_r(rad/s)');zlabel('am-
plitude');
title('\it\bfT\rm amp-freq property');

function[X,R,C]=toPreProcess(imgn)
imgd=imread(imgn);imginfo=imfinfo(imgn);
ifndims(imgd)==3
    X=uint8(0.30*imgd(:,:,1)+0.59*imgd(:,:,2)+0.11*imgd(:,:,3));
else
    X=imgd(:,:);
end
C=imginfo.Width;R=imginfo.Height;
% XFT=fftshift(abs(fft2(X,R,C)));
% XFT=(XFT-min(min(XFT)))/(max(max(XFT))-min(min(XFT)))*256;
% figure,imshow(X),figure,imshow(XFT);

function toINP(T,X,R,C)
n=(size(T,2)-1)/2;XX=X;XX(1:2:end,2:2:end)=0;
XX(2:2:end,1:2:end)=0;Xout=double(toEXP(XX,R,C,n));
Xout1=double(Xout);k=2;
```

```
if size(T,1) = =1
    for r =n +1:R +n
        for c =n +k:2:C +n
            Xout1(r ,c) =sum(T.* double(Xout(r,c -n:c +n)));
        end
        k =3 -k;
    end
else
    for r =n +1:R +n
        for c =n +k:2:C +n
            Xout1(r,c) =sum(sum(T.* double(Xout(r -n:r +n,c -n:c +n))));
        end
        k =3 -k;
    end
end

Y =(Xout1(n +1:R +n,n +1:C +n));
% figure ,imshow(X);figure,imshow(uint8(Y));

ERR =sum(sum((  Y -double(X)).^2))/numel(X);
disp('PSNR');10* log10(255^2/ERR)

functionXout =toEXP(xin,R,C,n)
lrud =fliplr(flipud(xin));ud =flipud(xin);lr =fliplr(xin);
xexp =[lrud ,ud,lrud;lr,xin,lr;lrud,ud,lrud];
Xout =xexp(R -n +1:2* R +n,C -n +1:2* C +n);
```

8.7　小结

本章首先介绍了模板设计的传统方法，对基于沃尔什变换、傅里叶变换和离散余弦变换的1/2、1/3、1/4 和1/8 内插模板提供了通解；其次，分别从列率谱和频谱方面分析图像1/2 抽取时谱调制规律，结合模板处理原理提出了重叠模板的概念，此模板在图像处理时具有最大程度重叠的特征，满足全相位处理的规律；第三，整理了基于DCT 的全相位模板设计原理（即Ⅰ型全相位模板），以第5 章内容为基础并集合模板概念提出了基于DFT/DWT/DCT 的无窗、单窗和双窗II 型全相位模板，从变换域角度进行低通滤波，因此具有小波处理特征，而且可以选择不同类型基础使模板构造自由度更大；最后，设计了2∶1 下取样的一维和二维信号的内插重构，对传统方法和全相位方法共27 种结果进行了对比，全相位内插模板处理效果普遍优于传统方法3dB（小波除外）。

处理信号谱特征、模板维度和窗函数是设计全相位模板需要考虑的 3 个重要因素，高频分量很少的信号满足上面设计模板时的假设条件，效果比较理想。但对于具有丰富高频的信号，传输特性的设置应该重新考虑。此外，窗函数的选择重新调整了模板系数，具有进一步低通处理效果，显然对高频会造成损伤，上面实验结果得到证实。如何根据期望目标特征量自适应地设计高性能模板是值得研究的方向。

第 9 章 基于W-O的自适应信号处理

9.1 引言

自适应滤波技术应用范围很广，不同的领域、不同的应用，自适应滤波技术也不尽相同。滤波器能够过滤或抑制输入信号中的干扰信号，提取有用信号。滤波器之所以能够滤波是因为它对不同频率的信号有不同的增益，因而能够将某些频率的信号放大，而另外的一些频率得到抑制。均衡器也可看作是一种滤波器，它主要用来消除信号通过传输系统后的线性失真。

要使滤波器能够最大限度地滤除干扰信号，就要设计最佳的频率响应特性和滤波参数。最佳滤波参数可根据输入信号的特性来确定，比如必须根据干扰信号和有用信号的频谱设计滤波器的频率响应，不可能设计一个对任何信号都是最佳的滤波器。当输入信号的特性未知或者随着时间缓慢变化时，用一般的滤波器不可能始终得到最佳的滤波效果，唯一的解决办法是引入自适应滤波器。自适应滤波器与普通滤波器有两个重要区别：

1）自适应滤波器的参数是可变的，它能够随着外界信号特性动态地改变参数，保持最佳滤波状态。

2）自适应滤波器除了普通滤波器的硬件设备外还有其软件部分，即自适应算法。自适应算法决定了自适应滤波器如何根据外界信号的变化来调整参数，自适应算法的好坏直接影响滤波的效果。

应用自适应滤波器主要有两种情形：

1）输入信号的特性是不变的，但是未知的。对于这种情形，最佳滤波参数是固定的。在这种情况下，要求自适应滤波器的参数尽快收敛到最佳滤波参数。一般把参数收敛过程称为“学习”过程。

2）输入信号的特性是缓慢变化的。这里所说的“缓慢变化”是相对于信号幅度变化而言的。在这种情况下，最佳滤波参数也是缓慢变化的，这就要求自适应滤波参数能尽快“反应”过来，跟随信号特性的变化而改变。一般称这个过程为“跟踪”过程。

自适应滤波器就是这样的处理器，它在输入信号特性未知或者输入信号特性

变化时，能够调整自己的参数，以满足某种最佳滤波准则的要求。显然，自适应滤波器对不同的应用，应采用不同的结构和不同的算法，只有这样，才能得到最佳的滤波效果。如此，自适应滤波器的研究工作也集中于两个方面：一是理论问题，一是综合问题。前者研究最佳的自适应算法，后者研究自适应滤波器的最佳实现方法。

9.2 自适应滤波器概述

自适应滤波器的结构如图 9-1 所示。自适应滤波器的硬件结构是可编程的，所以一般选用数字滤波器而不是模拟滤波器。

为了更容易地控制参数的变化，一般自适应滤波器是线性滤波器。自适应滤波器包括自适应时域滤波、自适应变换域、自适应频域滤波和自适应空域滤波（即自适应阵列）等，这里只讨论时域和变换域自适应滤波。

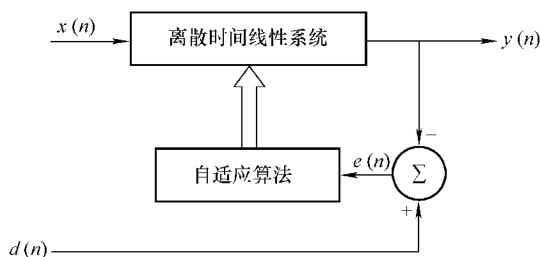


图 9-1 自适应滤波器原理框图

9.2.1 最佳滤波准则

从理论上讲，自适应滤波问题没有唯一的解。为了得到自适应滤波器及其应用系统，可以依据不同的准则实现不同的算法。常见的有最小均方误差准则（LMS），最小二乘准则、最大输出信噪比准则和统计检测准则等。最佳滤波准则和自适应滤波器关系密切，最佳滤波准则规定了与某种特性的信号对应的最佳参数，而这个最佳参数指出了自适应滤波器调整参数的方向。下面主要介绍 LMS 准则。

最小均方误差准则（Least Mean Square，LMS）就是要使输出信号和理想信号的误差平方的均值最小。图 9-1 中 $d(n)$ 为滤波器输出想要逼近的信号，$y(n)$ 为滤波器的输出信号，误差为 $e(n)=d(n)-y(n)$，LMS 准则就是要使 $E\{e^2(n)\}$ 达到最小。

9.2.2 自适应滤波器的应用

自适应滤波器的应用主要有以下 3 种形式：

1）如图 9-1 所示的联合过程估计。$x(n)$ 和 $d(n)$ 为已知，自适应滤波的目的是使 $y(n)$ 为 $d(n)$ 的最佳估计。这种应用形式常应用于自适应噪声对消。设

$s(n)$ 为有用信号，在 $s(n)$ 上叠加一个噪声信号 $n_0(n)$，$n_1(n)$ 是与 $n_0(n)$ 相关的另一个噪声信号，即

$$d(n)=s(n)+n_0(n),\ x(n)=n_1(n) \tag{9-1}$$

设 $s(n)$ 与 $n_0(n)$、$n_1(n)$ 不相关，则 $s(n)$ 与 $y(n)$ 不相关。使用 LMS 准则得：

$$\begin{aligned}E\{e^2(n)\}&=E\{[d(n)-y(n)]^2\}=E\{[s(n)+n_0(n)-y(n)]^2\}\\&=E\{s^2(n)\}+E\{[n_0(n)-y(n)]^2\}+2E\{s(n)[n_0(n)-y(n)]\}\end{aligned} \tag{9-2}$$

由于不相关，所以 $E\{s(n)[n_0(n)-y(n)]\}=0$。为了使 $E\{e^2(n)\}$ 最小，只能使 $E\{[n_0(n)-y(n)]^2\}$ 最小并且 $E\{e^2(n)\}\geqslant E\{s^2(n)\}$。即用 $y(n)$ 来估计 $d(n)$ 的最佳情况一般并不是 $E\{e^2(n)\}$ 为零，而是大于零的数。由 $E\{[n_0(n)-y(n)]^2\}$ 最小可知，$y(n)$ 实际逼近的是 $d(n)$ 中与 $y(n)$ 相关的量 $n_0(n)$ 而无法逼近与其无关的量 $s(n)$。又因为 $e(n)=s(n)+n_0(n)-y(n)$，所以 $e(n)$ 将趋近于有用信号 $s(n)$，这样成功地滤去了干扰信号而提取出有用信号。

2）自适应预测，如图 9-2 所示。自适应预测滤波器根据输入信号 $x(n)$ 的过去值 $x(n-1),x(n-2),\cdots,x(n-m)$ 来估计 $x(n)$，使估计值 $y(n)$ 为 $x(n)$ 的最佳估计。这种自适应滤波器常用于 ADPCM 语音编码。ADPCM 编码根据输入信号 $x(n)$ 的变化动态调整滤波器参数，使 $y(n)$ 为 $x(n)$ 的最佳估计，从而 $e(n)=x(n)-y(n)$ 很小，以减少量化比特数。接收端利用同样的自适应滤波器得到 $y(n)$，然后加上接收到的 $e(n)$ 即得 $x(n)$。

3）自适应参数估计，如图 9-3 所示。其中自适应滤波器的作用是估计输入信号的统计特性，这种自适应滤波器应用于声码器。自适应滤波得到声音信号的参量，通信中只传输参量。接收端根据这些参量恢复出声音信号。

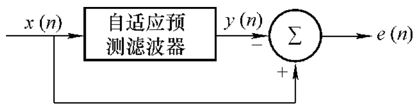


图 9-2　自适应预测

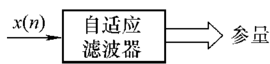


图 9-3　自适应参数估计

9.2.3　自适应滤波算法

衡量一种自适应算法的优劣一般有两个衡量标准，即参数收敛速度和收敛以后的稳态误差。然而这两个标准常常是相互矛盾的，当收敛速度加快，收敛精度就要降低，反之亦然。而且这两个标准必须和实际的应用场合结合起来分析，在不同的情况下对速度和精度的侧重不同。对于上面提到的自适应滤波器应用的第 1 种情形，输入信号特性未知但不随时间改变，由于大部分时间是处在收敛后的

状态下，所以速度是次要而精度比较重要。第 2 种情形中输入信号特性是变化的，所以收敛速度就很重要，最起码收敛速度应大于信号特性的变化速度。这种情况下自适应滤波器大部分时间处在未收敛状态，收敛后的精度就显得不太重要。

算法可以根据遵守的准则来分类，另外还有一种分法是分为分块算法和递归算法。分块算法是截取一段时间的输入信号（如从 $n-k$ 到 n）来估计最佳参数，而递归算法是一个闭环的算法，它不仅使用滤波器的输入信号，而且使用输出来估计未来参数，常见的 LMS 算法就是一种递归算法。由于递归算法的递归性，无限长时间以前的输入信号一直影响到当前的参数估计，于是新数据在估计中的比重就相对减少，这样当输入信号统计特性改变后，递归算法不能立即“反应”过来。

相反，分块算法只有块内（$n-k$ 到 n）的输入信号对参数的估计起作用，这样当输入信号统计特性改变以后，分块算法能很快地“反应”过来。但是分块算法常采用复杂的矩阵运算。这种算法常采用最小二乘准则即把块内的误差平方求和后要求其和最小，自适应算法的速度和精度的矛盾可以在分块算法中得到体现。当分块算法的块较大，则可以根据很长一段时间的输入来估计参数，于是估计就比较准确，但新数据在参数估计的影响下比重下降而使得跟踪速度下降。有一种特殊的算法，它的块是包括从 0 到 $N-1$ 的所有数据，这就是 RLS（Recursive Least Square）算法。

9.3 传统自适应算法

9.3.1 最小均方滤波算法

当在未知统计特性的环境下处理观测信号或数据以获得所期望的结果时，利用自适应滤波器可以得到令人满意的解，其性能远超过通用方法所设计的固定参数的滤波器。如果一个实用系统含有较多的自适应子系统，则它在自适应学习性能方面将有较高的自适应的先进性。自适应滤波器的两个重要组成部分中核心是算法，用来调整滤波子系统结构的参数或滤波系数。算法是指调节自适应滤波系数的步骤，以达到在所描述准则下的误差最小化。自适应滤波含有两个过程即自适应过程和滤波过程。前一过程的基本目标是调节滤波系数 $\theta(n)$ 使有意义的目标函数或代价函数 $\boldsymbol{F}(\cdot)$ 最小化，输出信号 $y(n)$ 逐步逼近所期望的参考信号 $d(n)$，由两者之间的估计误差 $e(n)$ 驱动某种算法对滤波系数进行调整，使滤波器处于最佳工作状态以实现滤波过程。所以，自适应过程是一个闭合的反馈环，算法决定了这个闭合环路的自适应过程所需要的时间。由于目标函数$\boldsymbol{F}(\cdot)$

是输入信号 $x(n)$、参考信号 $d(n)$ 及输出信号 $y(n)$ 的函数，因此目标函数必须具有以下两个性质：

1）非负性：$\boldsymbol{F}[x(n),d(n),y(n)]\geqslant 0\quad \forall x(n),d(n),y(n)$；

2）最佳性：$\boldsymbol{F}[x(n),d(n),y(n)]=0\quad d(n)=y(n)$。

在自适应过程中，自适应算法逐步逼近使目标函数 $\boldsymbol{F}(\cdot)$ 最小化，最终使 $y(n)$ 逼近于 $d(n)$，滤波参数或权系数 $\theta(n)$ 收敛于 θ_0 即自适应滤波系数的最优解。因此，自适应过程也是自适应滤波器的最佳线性估计过程，既要估计滤波器能实现期望信号 $d(n)$ 输出的整个过程，又要估计滤波权系数以进行有利于主要目标方向的调整。这些估计过程是时变的，这就是自适应滤波器需要有的自适应收敛过程。

最小均方横向自适应滤波结构如图9-4所示。图中 $x(n)$ 是输入信号，$y(n)$ 是输出信号，$d(n)$ 为 $y(n)$ 想要趋近的理想信号且为已知，$e(n)$ 为误差信号。假设两个条件：1）输入信号 $x(n)$ 和 $d(n)$ 是广义平稳的；2）$x(n)$ 和 $d(n)$ 的统计特性已知。

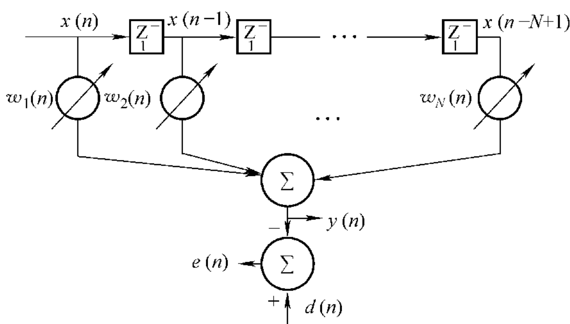


图9-4　用于联合参数估计的横式滤波结构

$x(n)$ 的自相关函数及与 $d(n)$ 的互相关函数为

$$\begin{aligned} r_{xx}(i) &= E\{x(n)x(n-i)\} \\ r_{xd}(i) &= E\{d(n)x(n-i)\} \end{aligned} \tag{9-3}$$

设加权矢量即滤波器系数矢量 $\boldsymbol{w}=[w_1\ w_2\cdots w_N]^{\mathrm{T}}$，滤波器在 n 时刻的输出为

$$y(n) = \sum_{i=1}^{N} w_i(n)x(n-i+1) = \boldsymbol{w}^{\mathrm{T}}\cdot\boldsymbol{x} \tag{9-4}$$

相应的误差为

$$e(n)=d(n)-y(n)=d(n)-\boldsymbol{w}\cdot\boldsymbol{x} \tag{9-5}$$

自适应滤波器控制机理是用误差序列 $e(n)$ 按照某种准则和算法对其系数 $\{w_i(n)\},(i=1,2,\cdots,N)$ 进行调节，最终使自适应滤波的目标（代价）函数最小化，达到最佳滤波状态。按照均方误差（MSE）准则所定义的目标函数是

$$\boldsymbol{F}[e(n)]=\xi(n)=E[e^2(n)]=E[d^2(n)-2d(n)y(n)+y^2(n)] \tag{9-6}$$

将式（9-4）代入到式（9-6）中并整理得：

$$\begin{aligned} \xi(n) &= E[d^2(n)]-2E[d(n)\boldsymbol{wx}]+E[\boldsymbol{w}^{\mathrm{T}}\boldsymbol{xx}^{\mathrm{T}}\boldsymbol{w}] \\ &= E[d^2(n)]-2\boldsymbol{w}^{\mathrm{T}}\boldsymbol{P}+\boldsymbol{w}^{\mathrm{T}}\boldsymbol{Rw}=\xi \end{aligned} \tag{9-7}$$

式（9-7）是均方误差性能函数，其中 $\boldsymbol{R}$ 是输入信号自相关矩阵。

$$\boldsymbol{R}=E[\boldsymbol{x}\boldsymbol{x}^{\mathrm{T}}]=\begin{bmatrix} r_{xx}(0) & r_{xx}(1) & \cdots & r_{xx}(N-1) \\ r_{xx}(1) & r_{xx}(0) & \cdots & r_{xx}(N-2) \\ \vdots & \vdots & \vdots & \vdots \\ r_{xx}(N-1) & r_{xx}(N-2) & \cdots & r_{xx}(0) \end{bmatrix} \tag{9-8}$$

$\boldsymbol{P}$ 是输入信号 $\boldsymbol{x}$ 与期望 $\boldsymbol{d}$ 的互相关矢量，定义如下：

$$\boldsymbol{P}=E[\boldsymbol{d}\cdot\boldsymbol{x}]=\begin{bmatrix} r_{xd}(0) \\ r_{xd}(1) \\ \vdots \\ r_{xd}(N-1) \end{bmatrix} \tag{9-9}$$

由式（9-7）可以看出，自适应滤波器的性能函数 $\xi(n)$ 是 $\boldsymbol{w}$ 的二次函数，且矩阵 $\boldsymbol{R}$ 对称且非负定，因此 $\xi(n)$ 有最小值。$\xi(n)$ 对 $\boldsymbol{w}$ 求梯度并使梯度等于零即可得到性能函数最小的最佳滤波系数为

$$\boldsymbol{w}_0=\boldsymbol{R}^{-1}\boldsymbol{P} \tag{9-10}$$

将上式结果代入到式（9-7）中并整理得到最小误差为

$$\xi_{\min}=E\{d^2(n)\}-\boldsymbol{w}_0\boldsymbol{P} \tag{9-11}$$

当矩阵 $\boldsymbol{R}$ 和 $\boldsymbol{P}$ 已知时，可由式（9-10）直接求解，把此解称为维纳解即最佳滤波系数值。因为 MSE 函数是滤波系数 $\boldsymbol{w}$ 的二次方程，由此形成一个多维的超抛物曲面，这好像一个碗状曲面又具有唯一的碗底最小点，通常称之为自适应滤波器的误差性能曲面。当滤波器工作在平稳随机过程的环境下，这个误差性能曲面就具有固定边缘的恒定形状。自适应滤波系数的起始值 $\{\boldsymbol{w}_i(0)\}$ $(i=1,2,\cdots,N)$ 是任意值，位于误差性能曲面上的某一点。经过自适应调节过程，使对应于滤波系数变化的点朝碗底最小点方向移动，最终到达碗底最小点得以实现最佳维纳滤波。

9.3.2 最陡下降法

最陡下降法是实现自适应滤波器最佳搜索的一种优化技术，它利用梯度信息分析自适应滤波性能和追踪最佳滤波状态。梯度矢量是由均方误差 $\xi(n)$ 的梯度来定义的，在多维超抛物曲面上任一点的梯度矢量是对应于均方误差 $\xi(n)$ 对滤波系数 $\boldsymbol{w}_i(n)$ 的一阶导数，由起始点或现在点变化到下一个点的滤波系数变化量正好是梯度矢量的负数。换句话说，自适应过程是在梯度矢量的负方向接连地校正滤波系数的，即在误差性能曲面的最陡下降方向移动和逐步校正滤波系数，最终到达均方误差为最小的碗底最小点，获得最佳滤波或次优工作状态。

令 $\nabla(n)$ 代表 n 时刻的 $N\times1$ 维梯度矢量，这里 N 等于滤波器滤波系数个数；$\boldsymbol{w}(n)$ 为自适应滤波器在 n 时刻的滤波系数或权矢量。按照最陡下降法调节

滤波系数，则在 $n+1$ 时刻的滤波系数或权矢量 $\boldsymbol{w}(n+1)$ 可以用下列简单的递归关系来计算：

$$\boldsymbol{w}(n+1)=\boldsymbol{w}(n)+\mu[-\nabla(n)] \tag{9-12}$$

式中，μ 是一个正实数，通常称它为自适应收敛系数或步长。由梯度的定义，$\nabla(n)$ 可以写成：

$$\begin{aligned}\nabla(n)&=\frac{\partial E[e^2(n)]}{\partial \boldsymbol{w}(n)}=\left[\frac{\partial\xi(n)}{\partial w_1(n)}\frac{\partial\xi(n)}{\partial w_2(n)}\cdots\frac{\partial\xi(n)}{\partial w_N(n)}\right]\\&=E\left[2e(n)\frac{\partial e(n)}{\partial \boldsymbol{w}(n)}=-2E[e(n)\boldsymbol{x}(n)]\right]\end{aligned} \tag{9-13}$$

当滤波系数为最佳值即维纳解时，梯度矢量 $\nabla(n)=E[e(n)\boldsymbol{x}(n)]=0$。这意味着误差信号与输入信号矢量的每一个分量都正交（即互不相关）。且由式（9-5）不难证明：

$$E[e(n)y(n)]=0 \tag{9-14}$$

结合式（9-7）和式（9-13），可求得梯度如下：

$$\nabla(n)=-2\boldsymbol{P}+2\boldsymbol{R}\boldsymbol{w}(n) \tag{9-15}$$

因此，在最陡下降法中，当相关矩阵 $\boldsymbol{R}$ 与互相关矢量 $\boldsymbol{P}$ 已知时，则由 $\boldsymbol{w}(n)$ 可以计算梯度矢量 $\nabla(n)$，然后将式（9-15）代入到式（9-12）中，可以计算滤波系数的更新值：

$$\boldsymbol{w}(n+1)=\boldsymbol{w}(n)+2\mu[\boldsymbol{P}-\boldsymbol{R}\boldsymbol{w}(n)] \tag{9-16}$$

式中，μ 是收敛因子。式（9-16）是最陡下降法的数学公式，由此可得到信号流图如图 9-5 所示。图中，$\boldsymbol{I}$ 是 N 阶单位阵。

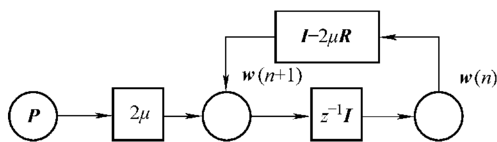


图 9-5　最陡下降算法的信号流图

9.3.3　最小均方 LMS 算法

最陡下降算法不需要知道误差特性曲面的先验知识，其算法就能收敛到最佳维纳解且与起始条件无关。但是最陡下降算法的主要限制是它需要准确测得每次迭代的梯度矢量，这妨碍了它的应用。为了减少计算复杂度和缩短自适应收敛时间，1960 年，美国斯坦福大学的 Widrow 等人提出了 LMS 算法，这是一种用瞬时值估计梯度矢量的方法，即

$$\hat{\nabla}(n)=\frac{\partial[e^2(n)]}{\partial \boldsymbol{w}(n)}=-2e(n)\boldsymbol{x}(n) \tag{9-17}$$

因为 $E[\hat{\nabla}(n)]=\nabla(n)$，所以这种瞬时估计法是无偏的。按照自适应滤波系数矢量的变化与梯度矢量方向之间的关系，可写出 LMS 算法的公式如下：

$$\hat{\boldsymbol{w}}(n+1)=\hat{\boldsymbol{w}}(n)+\mu[-\hat{\nabla}(n)]=\hat{\boldsymbol{w}}(n)+2\mu e(n)\boldsymbol{x}(n) \tag{9-18}$$

把式（9-5）代入到式（9-18）中得：

$$\begin{aligned}\hat{\boldsymbol{w}}(n+1)&=\hat{\boldsymbol{w}}(n)+2\mu\boldsymbol{x}(n)[d(n)-\hat{\boldsymbol{w}}^{\mathrm{H}}(n)\boldsymbol{x}(n)]\\&=[\boldsymbol{I}-2\mu\boldsymbol{x}(n)\boldsymbol{x}^{\mathrm{H}}(n)]\hat{\boldsymbol{w}}(n)+2\mu\boldsymbol{x}(n)d(n)\end{aligned} \tag{9-19}$$

由式（9-19）可以得到自适应 LMS 算法的信号流图，如图 9-6 所示。这是一个具有反馈形式的模型，如同最陡下降法，利用时间 $n=0$ 的滤波系数矢量为任意的起始值 $\boldsymbol{w}$（0），然后开始 LMS 算法的计算，其步骤如下：

1）由现在时刻 n 的矢量估值 $\hat{\boldsymbol{w}}$（n）、$\boldsymbol{x}$（n）以及期望信号 d（n）按照式（9-5）计算误差信号 e（n）。

2）利用递归算法计算滤波器系数矢量的更新估值。

3）将时间指数 n 增加 1，重回到步骤 1）执行直到达稳态为止。

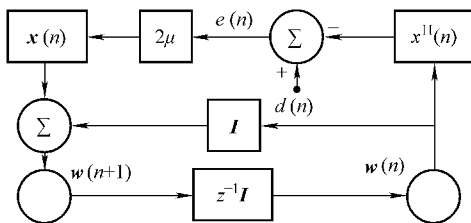


图 9-6　自适应 LMS 算法信号流图

由此可见，自适应 LMS 算法简单，它既不需要计算输入信号的相关函数，也不要求矩阵可逆。但是，由于 LMS 算法采用梯度矢量的瞬时估计值，它有较大的方差以致不能获得最优滤波性能。下面从收敛性、学习曲线和失调 3 个方面分析 LMS 算法的收敛性能。

1. 自适应收敛性

自适应滤波器系数矢量的起始值 $\boldsymbol{w}(0)$ 是任意常数，应用 LMS 算法调节滤波系数具有随机性而使矢量 $\boldsymbol{w}(n)$ 带来非平稳过程，通常为了简化 LMS 算法的统计分析，往往假设算法连续迭代之间存在以下的充分条件：

1）每个输入信号样本矢量 $\boldsymbol{x}(n)$ 与其过去全部样本矢量 $\boldsymbol{x}(k)$，$k=0,1,\cdots,n-1$ 统计独立且不相关，即 $E[\boldsymbol{x}(n)\boldsymbol{x}^{\mathrm{H}}(k)]=0$。

2）每个输入信号样本矢量 $\boldsymbol{x}(n)$ 与全部过去的期望信号 $\boldsymbol{d}(k)$，$k=0,1,\cdots,n-1$ 统计独立且不相关，即 $E[\boldsymbol{x}(n)\boldsymbol{d}^{\mathrm{H}}(k)]=0$。

3）期望信号样本 $\boldsymbol{d}(n)$ 依赖于输入过程样本矢量 $\boldsymbol{x}$（n），但全部过去的期望信号样本统计独立。

4）滤波器抽头输入信号矢量 $\boldsymbol{x}(n)$ 与期望信号 $\boldsymbol{d}(n)$ 包含着全部 n 个共同的高斯分布随机变量。

在上述假设的前提下，自适应滤波器在 $n+1$ 时刻的滤波系数矢量 $\hat{\boldsymbol{w}}(n+1)$ 依赖于 3 个输入：①输入过程的过去样本矢量 $\boldsymbol{x}(k)$，$k=0,1,\cdots,n-1$；②期

望信号的以前样本值 $\boldsymbol{d}(k)$，$k=0$，1，…，$n-1$；③滤波系数矢量的起始值$\hat{\boldsymbol{w}}(0)$。

对式（9-19）两端求数学期望，可得到：

$$\begin{aligned} E[\hat{\boldsymbol{w}}(n+1)] &= E[\hat{\boldsymbol{w}}(n)] + 2\mu E\{\boldsymbol{x}(n)[d(n)-\boldsymbol{x}^{\mathrm{T}}(n)\boldsymbol{w}(n)]\} \\ &= (I-2\mu\boldsymbol{R}_{\mathrm{xx}})E[\hat{\boldsymbol{w}}(n)] + 2\mu\boldsymbol{R}_{\mathrm{xd}} \end{aligned} \tag{9-20}$$

当 $k=0$ 时，利用式（9-20）可求得：

$$E[\hat{\boldsymbol{w}}(1)] = (I-2\mu\boldsymbol{R}_{\mathrm{xx}})E[\hat{\boldsymbol{w}}(0)] + 2\mu\boldsymbol{R}_{\mathrm{xd}} \tag{9-21}$$

当 $k=1$ 时，利用式（9-21）的结果可得：

$$E[\hat{\boldsymbol{w}}(1)] = (I-2\mu\boldsymbol{R}_{\mathrm{xx}})^2E[\hat{\boldsymbol{w}}(0)] + 2\mu\sum_{i=0}^{1}(I-2\mu\boldsymbol{R}_{\mathrm{xx}})^i\boldsymbol{R}_{\mathrm{xd}} \tag{9-22}$$

依次递推就得到通式如下：

$$E[\hat{\boldsymbol{w}}(k+1)] = (I-2\mu\boldsymbol{R}_{\mathrm{xx}})^{k+1}E[\hat{\boldsymbol{w}}(0)] + 2\mu\sum_{i=0}^{k}(I-2\mu\boldsymbol{R}_{\mathrm{xx}})^i\boldsymbol{R}_{\mathrm{xd}} \tag{9-23}$$

式中，R_{xx}是 Hermite 矩阵，其特征值分解式可写成如下：

$$\boldsymbol{R}_{\mathrm{xx}} = \boldsymbol{Q}\sum\boldsymbol{Q}^{\mathrm{T}} = \boldsymbol{Q}\sum\boldsymbol{Q}^{-1} \tag{9-24}$$

式中，$\boldsymbol{Q}$ 是正定特征矩阵；$\sum = =\mathrm{diag}(\lambda_0,\lambda_1,\cdots,\lambda_M)$是特征值对角阵，$\lambda_i$ 是R_{xx}的特征值。把式（9-24）代入到式（9-23）并注意到 $E[\hat{\boldsymbol{w}}(0)]=\hat{\boldsymbol{w}}(0)$，所以得：

$$\begin{aligned} E[\hat{\boldsymbol{w}}(k+1)] &= (I-2\mu\boldsymbol{Q}\sum\boldsymbol{Q}^{-1})^{k+1}E[\hat{\boldsymbol{w}}(0)] + 2\mu\sum_{i=0}^{k}(I-2\mu\boldsymbol{Q}\sum\boldsymbol{Q}^{-1})^i\boldsymbol{R}_{\mathrm{xd}} \\ &= \boldsymbol{Q}(\boldsymbol{I}-2\mu\sum)^{k+1}\boldsymbol{Q}^{-1} + 2\mu\boldsymbol{Q}\sum_{i=0}^{k}(\boldsymbol{I}-2\mu\sum)^i\boldsymbol{Q}^{-1}\boldsymbol{R}_{\mathrm{xd}} \end{aligned} \tag{9-25}$$

显然，当迭代次数 k 增加到很大时，式（9-25）收敛的条件是：$\boldsymbol{I}-2\mu\sum$ 的对角线元素均大于 0 且小于 1，即要求满足：

$$0<1-2\mu\lambda_{\max}<1\Rightarrow\lambda_{\max}<\frac{1}{\mu} \tag{9-26}$$

式中，$\lambda_{\max}$是 R_{xx}的最大特征值；μ 是收敛因子或步长。把最大特征值与最小特征值的比值 $d=\lambda_{\max}/\lambda_{\min}$称为谱动态范围，它决定着收敛速度，$d$ 越大收敛速度越小。降低 d 的方法之一是产生正交数据。在条件（9-26）满足的条件下，式（9-25）等于：

$$\begin{aligned} E[\hat{\boldsymbol{w}}(k+1)] &= 0 + 2\mu\boldsymbol{Q}\left(2\mu\sum\right)^{-1}\boldsymbol{Q}^{-1}\boldsymbol{R}_{\mathrm{xd}} = \boldsymbol{Q}\left(\sum\right)^{-1}\boldsymbol{Q}^{-1}\boldsymbol{R}_{\mathrm{xd}} \\ &= \boldsymbol{R}_{\mathrm{xx}}^{-1}\boldsymbol{R}_{\mathrm{xd}} = \boldsymbol{w}_{\mathrm{opt}} \end{aligned} \tag{9-27}$$

2. 平均 **MSE**——学习曲线

最陡下降法每次迭代都要精确计算梯度矢量，使自适应横向滤波器权矢量或

滤波系数矢量 $\boldsymbol{w}(n)$ 能达到最佳维纳解 $\boldsymbol{w}(0)$，这时滤波器 MSE 为最小，即：

$$\xi_{\min} = \sigma_d^2 - \boldsymbol{w}^{\mathrm{T}}\boldsymbol{P} \tag{9-28}$$

式中，σ_d^2 是期望信号 $\boldsymbol{d}(n)$ 的方差。学习曲线定义为均方误差随迭代计算次数 n 的变化关系，即

$$\xi(n) = \xi_{\min} + \sum_{i=1}^{N} \lambda_i (1 - \mu\lambda_i)^{2n} v_i^2(0) \tag{9-29}$$

式中，v_i 是 $\boldsymbol{w}(n)$ 正交旋转后对应矢量中的分量。式中每个指数项对应算法的固有模式，模式的数目等于滤波加权数。显而易见，由于上式中 $1-\mu\lambda_i<1$，故当 $n\to\infty$ 时最陡下降算法的均方误差 $\xi(\infty)=\xi_{\min}$。但 LMS 算法用瞬时值估计梯度存在误差的噪声估计，结果使滤波器权矢量估值 $\hat{\boldsymbol{w}}(n)$ 只能接近维纳解，这意味着滤波均方误差 $\xi(n)$ 随着迭代次数 n 的增加而出现小波动地减小，最后 $\xi(\infty)$ 不是等于 $\xi_{\min}$ 而是稍大于其值，如图 9-7 所示。如果步长选得越小，则噪化指数衰减曲线上的波动幅度将越小，即学习曲线的平滑度越好。

但是，对于自适应横向滤波器总体来说，假设每个滤波器 LMS 算法用相同的步长和同样的起始系数矢量，并从同一统计群体随机地选取各个平稳的各态历经的输入信号，由此计算自适应滤波器总体平均学习曲线，如图 9-8 所示。这是一条平滑的总体平均学习曲线，通常它是由 50 ~ 200 个单独 LMS 算法的结果加以平均而得到。显然，可以用 $E[\xi(n)]$ 表示的平均 RMS 来描述 LMS 算法的动态性质。

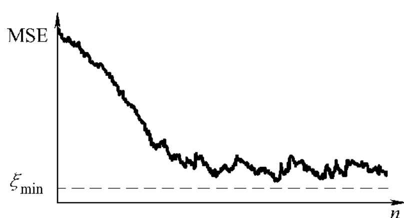


图 9-7　单条学习曲线

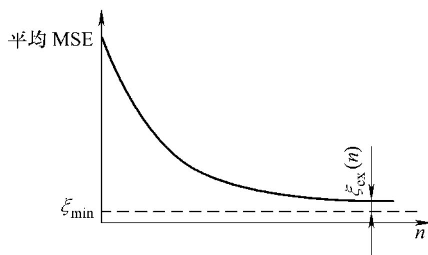


图 9-8　总体平均学习曲线

3. 失调

在自适应滤波器中，失调 χ 是衡量其滤波性能的一个技术指标，它被定义为总体平均超量均方误差值 $\xi_{ex}(\infty)$ 与最小均方误差值 $\xi_{\min}$ 之比，即

$$\chi = \frac{E[\xi_{ex}(\infty)]}{\xi_{\min}} \tag{9-30}$$

可推导得出：

$$\xi_{ex}(\infty) = \frac{\mu\xi_{\min}\sum_{i=1}^{N}\lambda_i}{1-\mu\sum_{i=1}^{N}\lambda_i} \tag{9-31}$$

将式（9-31）结果代入到式（9-30）中得到：

$$\chi = \frac{\mu \sum_{i=1}^{N} \lambda_i}{1 - \mu \sum_{i=1}^{N} \lambda_i} \tag{9-32}$$

通常步长 μ 值很小，因此，失调又可近似表示为

$$\chi = \mu \sum_{i=1}^{N} \lambda_i \tag{9-33}$$

由此可以看出，自适应滤波器 LMS 算法的稳态失调与步长成正比。当步长很小时，失调也降低，但收敛时间也会变长，因此，失调与自适应收敛之间存在着矛盾。

4. 缩短收敛过程的方法

由 LMS 算法的迭代公式（9-18）可以看出，为了缩短收敛过程，可以从 3 个方面进行设计：

1）采用不同的梯度估值 $\nabla(n)$，如 LMS 牛顿法，它估计 $\nabla(n)$ 时采用了输入矢量相关函数的估值，使得收敛速度大大优于上述经典的 LMS 算法，因为它在迭代过程中采用了更多的有关输入信号矢量的信息。

2）对收敛因子步长 μ 选用不同方法。步长 μ 的大小决定着算法的收敛速度和达到稳态的失调量的大小。对于常数 μ 值来说，收敛速度和失调量是一对矛盾，要想得到较快的收敛速度可以选用大的 μ 值，这将导致大的失调量；如果要满足失调量的要求，则收敛速度将受到制约。因此，人们研究了采用变步长的方法来克服这一矛盾。自适应过程开始时，取用较大的步长以保证较快的收敛速度，然后让 μ 值逐渐减小，以保证收敛后得到较小的失调量。如归一化 LMS 算法、时域正交化 LMS 算法等都是采用变步长方法的高效自适应算法。

3）采用变换域分块处理技术。对由滤波器权系数矢量调整的修正项中的乘积用变换域快速算法与分块处理技术可以大大减少计算量，且能改善收敛特性，如频域 LMS 算法、分块 LMS 算法等。

9.4　时域加权重叠 LMS 算法（WO-LMS）

由第 4 章中内容可知，任何正交变换域 ap 在时域中都可以等效为一个 FIR 滤波器，与传统滤波器不同的是，在输入端不是原始信号的直接输入而是首先对它进行加权后再重叠，得到的结果作为滤波器的实际输入。将这种对信号的加权重叠预处理思想运用于以 LMS 为准则的自适应滤波算法中，提出加权重叠 LMS 自适应滤波算法（WO-LMS）。与 LMS 相比，它提高了收敛速度同时使稳态失调

大为降低。

结合横向 LMS 滤波器结构，得到相应的 WO-LMS 系统如图 9-9 所示。

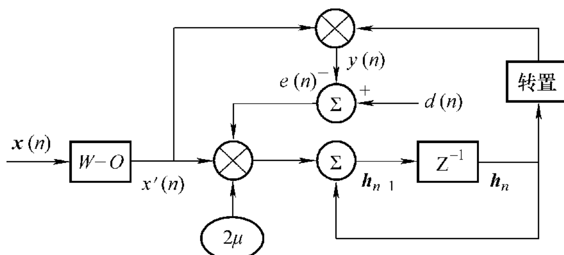


图 9-9　WO-LMS 自适应滤波器信号流图

图 9-9 中，$x(n)=[x(n-N+1)\cdots x(n-1)x(n)x(n+1)\cdots x(n+N-1)]^{\mathrm{T}}$ 是系统在 n 时刻的输入，N 是系统权的个数。$\boldsymbol{x}'(n)$ 是 $x(n)$ 的加窗重叠处理。$y(n)$ 是系统在 n 时刻的输出，$d(n)$ 是期望响应，$e(n)$ 是 n 时刻的误差，$\boldsymbol{h}$ 是系统冲击响应，μ 是步长。由图 9-9 可以得到如下关系式：

$$\boldsymbol{x}'(n)=\boldsymbol{O}\cdot\boldsymbol{W}\cdot\boldsymbol{x}(n) \tag{9-34}$$

$$y(n)=\boldsymbol{h}_n^{\mathrm{T}}\boldsymbol{x}'(n) \tag{9-35}$$

$$e(n)=d(n)-y(n) \tag{9-36}$$

$$\boldsymbol{h}_{n+1}=\boldsymbol{h}_n+2\mu e(n)\boldsymbol{x}'(n) \tag{9-37}$$

式中，$\boldsymbol{O}$ 是重叠矩阵；$\boldsymbol{W}$ 是加权矩阵，显然加权矩阵等于延迟窗 $\boldsymbol{C}$ 构成的对角阵。将式（9-34）分别代入到式（9-35）和式（9-37）中并整理得：

$$y(n)=(\boldsymbol{E}\cdot\boldsymbol{h}_n)^{\mathrm{T}}\boldsymbol{x}(n) \tag{9-38}$$

$$\boldsymbol{h}_{n+1}=\boldsymbol{h}_n+2\mu e(n)\boldsymbol{E}^{\mathrm{T}}\cdot\boldsymbol{x}(n) \tag{9-39}$$

$\boldsymbol{O}$ 和 $\boldsymbol{W}$ 的乘积即为全相位转移矩阵 $\boldsymbol{E}$ 的转置，即：

$$\boldsymbol{E}^{\mathrm{T}}=\boldsymbol{O}\cdot\boldsymbol{W}=\frac{1}{N}\begin{bmatrix} 0 & 0 & \cdots & 0 & N & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & N-1 & 0 & 0 & 0 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots & 0 & \vdots & \vdots & \ddots & 0 \\ 0 & 2 & \cdots & 0 & 0 & 0 & N-2 & \cdots & 0 \\ 1 & 0 & \cdots & 0 & 0 & N-1 & 0 & 0 & 0 \end{bmatrix} \tag{9-40}$$

结合式（9-35）、式（9-36）和式（9-38），可求得均方误差为

$$E[e^2(n)]=E[(d(n)-\boldsymbol{h}_n^{\mathrm{T}}\boldsymbol{E}^{\mathrm{T}}\boldsymbol{x}(n))^2] \tag{9-41}$$

将式（9-41）展开整理得：

$$E[e^2(n)]=E[y^2(n)]-2\,\boldsymbol{h}_n^{\mathrm{T}}\boldsymbol{E}^{\mathrm{T}}\boldsymbol{P}+\boldsymbol{h}_n^{\mathrm{T}}\boldsymbol{E}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{E}\,\boldsymbol{h}_n \tag{9-42}$$

式中，$\boldsymbol{P}=E[d(n)\boldsymbol{x}(n)]$；$\boldsymbol{R}=E[\boldsymbol{x}(n)\boldsymbol{x}^{\mathrm{T}}(n)]$。上式对 $\boldsymbol{h}_n$ 求偏导等于 0，求出

使均方误差最小的冲击响应 $\boldsymbol{h}^*$ 满足：

$$\boldsymbol{E}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{E}\,\boldsymbol{h}^* = \boldsymbol{E}^{\mathrm{T}}\boldsymbol{P} \tag{9-43}$$

用响应误差向量 $\Delta\boldsymbol{h}_n = \boldsymbol{h}_n - \boldsymbol{h}^*$ 代入式（9-42），并结合式（9-36）和式（9-38）得到：

$$\begin{aligned}\boldsymbol{h}_{n+1} &= [\boldsymbol{I} - 2\mu\cdot\boldsymbol{E}^{\mathrm{T}}\boldsymbol{x}(n)\boldsymbol{x}^{\mathrm{T}}(n)\boldsymbol{E}](\Delta\boldsymbol{h}_n + \boldsymbol{h}^*) + 2\mu\cdot\boldsymbol{E}^{\mathrm{T}}\boldsymbol{x}(n)d(n)\\ &= [\boldsymbol{I} - 2\mu\cdot\boldsymbol{E}^{\mathrm{T}}\boldsymbol{x}(n)\boldsymbol{x}^{\mathrm{T}}(n)\boldsymbol{E}]\Delta\boldsymbol{h}_n + \boldsymbol{h}^* + 2\mu\cdot\boldsymbol{E}^{\mathrm{T}}[\boldsymbol{x}(n)d(n) - \boldsymbol{x}(n)\boldsymbol{x}^{\mathrm{T}}(n)\boldsymbol{E}\,\boldsymbol{h}^*]\\ &\Rightarrow \Delta\boldsymbol{h}_{n+1} = [\boldsymbol{I} - 2\mu\cdot\boldsymbol{E}^{\mathrm{T}}\boldsymbol{x}(n)\boldsymbol{x}^{\mathrm{T}}(n)\boldsymbol{E}]\Delta\boldsymbol{h}_n + 2\mu\cdot\boldsymbol{E}^{\mathrm{T}}[\boldsymbol{x}(n)d(n) - \boldsymbol{x}(n)\boldsymbol{x}^{\mathrm{T}}(n)\boldsymbol{E}\,\boldsymbol{h}^*]\end{aligned} \tag{9-44}$$

式中，$\Delta\boldsymbol{h}_{n+1} = \boldsymbol{h}_{n+1} - \boldsymbol{h}^*$ 表示响应误差矢量的更新值。为了分析方便，假定 $\Delta\boldsymbol{h}$ 和 $\boldsymbol{x}(n)$ 相互独立。对式（9-44）两边取数学期望得到：

$$\begin{aligned}E[\Delta\boldsymbol{h}_{n+1}] &= E\{[\boldsymbol{I} - 2\mu\cdot\boldsymbol{E}^{\mathrm{T}}\boldsymbol{x}(n)\boldsymbol{x}^{\mathrm{T}}(n)\boldsymbol{E}]\Delta\boldsymbol{h}_n\} + 2\mu\cdot E\{\boldsymbol{E}^{\mathrm{T}}[\boldsymbol{x}(n)d(n) - \boldsymbol{x}(n)\boldsymbol{x}^{\mathrm{T}}(n)\boldsymbol{E}\,\boldsymbol{h}^*]\}\\ &= (\boldsymbol{I} - 2\mu\cdot\boldsymbol{E}^{\mathrm{T}}E[\boldsymbol{x}(n)\boldsymbol{x}^{\mathrm{T}}(n)]\boldsymbol{E})E[\Delta\boldsymbol{h}_n] + 2\mu\cdot\boldsymbol{E}^{\mathrm{T}}E[\boldsymbol{x}(n)d(n)] - 2\mu\cdot\boldsymbol{E}^{\mathrm{T}}E\\ &\quad [\boldsymbol{x}(n)\boldsymbol{x}^{\mathrm{T}}(n)]\boldsymbol{E}\,\boldsymbol{h}^*\\ &= (\boldsymbol{I} - 2\mu\cdot\boldsymbol{E}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{E})E[\Delta\boldsymbol{h}_n] + 2\mu(\boldsymbol{E}^{\mathrm{T}}\boldsymbol{P} - \boldsymbol{E}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{E}\,\boldsymbol{h}^*)\end{aligned} \tag{9-45}$$

由式（9-43）可知式（9-45）的第二项等于 0 矩阵，所以又可以简化为

$$E[\Delta\boldsymbol{h}_{n+1}] = (\boldsymbol{I} - 2\mu\cdot\boldsymbol{D})E[\Delta\boldsymbol{h}_n] \tag{9-46}$$

式中，$\boldsymbol{D} = \boldsymbol{E}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{E}$。显然，$\boldsymbol{D}$ 是 Hermite 型矩阵。由 LMS 收敛准则知，当步长满足下式时式（9-46）收敛：

$$0 < \mu < 1/\lambda_{\max} \tag{9-47}$$

式中，$\lambda_{\max}$是 $\boldsymbol{D}$ 的最大特征值。由矩阵理论可知：

$$\mathrm{tr}(\boldsymbol{D}) = \mathrm{tr}(\boldsymbol{E}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{E}) \leqslant \mathrm{tr}(\boldsymbol{E}^{\mathrm{T}}\boldsymbol{E})\,\mathrm{tr}(\boldsymbol{R}) \tag{9-48}$$

由全相位转移矩阵定义，可知：

$$\boldsymbol{E}^{T}\boldsymbol{E} = \frac{1}{N^2}\mathrm{diag}[N^2\ (N-1)^2 + 1 \cdots 2^2 + (N-2)^2 1 + (N-1)^2] \tag{9-49}$$

结合式（9-48）和式（9-49）得：

$$\mathrm{tr}(\boldsymbol{D}) \leqslant \frac{2N^2+1}{3N}\mathrm{tr}(\boldsymbol{R}) \tag{9-50}$$

将式（9-50）和式（9-47）结合得：

$$0 < \mu \leqslant \frac{3N}{2N^2+1}\frac{1}{\mathrm{tr}(\boldsymbol{R})} \leqslant \frac{1}{\mathrm{tr}(\boldsymbol{D})} < \frac{1}{\lambda_{\max}} \tag{9-51}$$

同理可以求出 $\mathrm{ap}^{1\mathrm{sd}}$对应自适应滤波算法收敛的步长范围。当 N 相当大时，进一步放小不等式得到 3 种步长的范围依次为（各种窗经过归一化处理）：

$$\mathrm{ap}^{1\mathrm{n}}:\ 0 < \mu \leqslant \frac{1}{N\mathrm{tr}(\boldsymbol{R})}\qquad \mathrm{ap}^{1\mathrm{s}}:\ 0 < \mu \leqslant \frac{1}{N^2\mathrm{tr}(\boldsymbol{R})}\qquad \mathrm{ap}^{1\mathrm{d}}:\ 0 < \mu \leqslant \frac{1}{N^3\mathrm{tr}(\boldsymbol{R})} \tag{9-52}$$

上面就是 WO-LMS 算法收敛的条件。

值得注意的是，LMS 算法的收敛性能依赖输入信号的谱动态范围。谱动态范围越大的收敛性越差，越小则收敛性能越好。WO-LMS 算法由于对输入信号进行了加权重叠处理，使算法的收敛性得到改善。

由上面的分析可知，加权重叠处理前后系统的收敛性能分别取决于相关矩阵 $\boldsymbol{R}_{xx}$ 和 $\boldsymbol{R}_{x'x'}$ 的特征值分布。令 tr（$\boldsymbol{M}$）与 det（$\boldsymbol{M}$）分别表示 N 阶方阵 $\boldsymbol{M}$ 的迹和行列式值，则其最大与最小特征值 λ_{max}、λ_{min} 有如下关系：

$$\lambda_{max} \leqslant \mathrm{tr}(\boldsymbol{M})\lambda_{min} \geqslant \det(\boldsymbol{M})N>2 \tag{9-53}$$

这样，有

$$\gamma(\boldsymbol{M})=\frac{\mathrm{tr}(\boldsymbol{M})}{\det(\boldsymbol{M})} \tag{9-54}$$

可以作为矩阵 $\boldsymbol{M}$ 谱动态范围的上限。利用上述结果由式（9-38）可知，

$$\gamma(\boldsymbol{R}_{x'x'})=\frac{\mathrm{tr}(\boldsymbol{E}^{\mathrm{T}}\boldsymbol{R}_{xx}\boldsymbol{E})}{\det(\boldsymbol{E}^{\mathrm{T}}\boldsymbol{R}_{xx}\boldsymbol{E})} \tag{9-55}$$

根据矩阵理论的一般知识，有

$$\mathrm{tr}(\boldsymbol{E}^{\mathrm{T}}\boldsymbol{R}_{xx}\boldsymbol{E}) \leqslant \mathrm{tr}(\boldsymbol{E}^{\mathrm{T}}\boldsymbol{E})\,\mathrm{tr}(\boldsymbol{R}_{xx}) \tag{9-56}$$

对于 ap^{1nsd} 的转移矩阵 $\boldsymbol{E}$ 具有“双对角”形状，对角线上的元素等于卷积窗 $\boldsymbol{C}$ 的对应元素。以 N 阶 ap^{1n} 为例，令 $\boldsymbol{E}=[\boldsymbol{E}_1,\boldsymbol{E}_2]$，其中：

$$\boldsymbol{E}_1=\frac{1}{N}\begin{bmatrix} 0 & 0 & \cdots & N \\ 0 & 0 & 2 & 0 \\ \vdots & \iddots & \vdots & \vdots \\ 1 & 0 & \cdots & 0 \end{bmatrix}_{N\times N} \quad \boldsymbol{E}_2=\frac{1}{N}\begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & N-2 & \cdots & 0 \\ N-1 & 0 & \cdots & 0 \end{bmatrix}_{N\times(N-1)} \tag{9-57}$$

设矢量 $\boldsymbol{x}(n)$ 的前 N 个输入 $\boldsymbol{x}_1(n)=[x(n-N+1)\cdots x(n-1)x(n)]^{\mathrm{T}}$ 和后 $N-1$ 个输入 $\boldsymbol{x}_2(n)=[x(n-1)x(n-2)\cdots x(n-N+1)]^{\mathrm{T}}$ 的自相关和互相关分别为 $\boldsymbol{r}_{11}$、$\boldsymbol{r}_{22}$ 与 $\boldsymbol{r}_{12}$、$\boldsymbol{r}_{21}$，则

$$\boldsymbol{R}_{xx}=\begin{bmatrix} \boldsymbol{r}_{11} & \boldsymbol{r}_{12} \\ \boldsymbol{r}_{21} & \boldsymbol{r}_{22} \end{bmatrix} \tag{9-58}$$

按照 LMS 算法给出的充分条件，$\boldsymbol{x}_1$ 和 $\boldsymbol{x}_2$ 的相关性很小即认为 $\boldsymbol{r}_2=\boldsymbol{r}_{21}^{\mathrm{T}}\approx 0$。所以

$$\det(\boldsymbol{R}_{xx})\approx\det(\boldsymbol{r}_{11})\det(\boldsymbol{r}_{22}) \tag{9-59}$$

加权重叠信号 $\boldsymbol{x}'$ 的自相关等于：

$$\begin{aligned}\boldsymbol{R}_{x'x'}&=[\boldsymbol{E}_1^{\mathrm{T}} \quad \boldsymbol{E}_2^{\mathrm{T}}]\begin{bmatrix} \boldsymbol{r}_{11} & \boldsymbol{r}_{12} \\ \boldsymbol{r}_{21} & \boldsymbol{r}_{22} \end{bmatrix}\begin{bmatrix} \boldsymbol{E}_1 \\ \boldsymbol{E}_2 \end{bmatrix} \\ &=\boldsymbol{E}_1^{\mathrm{T}}\boldsymbol{r}_{11}\boldsymbol{E}_1+\boldsymbol{E}_2^{\mathrm{T}}\boldsymbol{r}_{22}\boldsymbol{E}_2+\boldsymbol{E}_2^{\mathrm{T}}\boldsymbol{r}_{21}\boldsymbol{E}_1+\boldsymbol{E}_{1\ 1}^{\mathrm{T}}\boldsymbol{r}_{12}\boldsymbol{E}_2 \\ &\approx\boldsymbol{E}_1^{\mathrm{T}}\boldsymbol{r}_{11}\boldsymbol{E}_1+\boldsymbol{E}_2^{\mathrm{T}}\boldsymbol{r}_{22}\boldsymbol{E}_2\end{aligned} \tag{9-60}$$

又由于 $\boldsymbol{r}_{11}$、$\boldsymbol{r}_{22}$ 均为正定矩阵且 $\boldsymbol{E}_2$ 为行线性相关，因此

$$\begin{aligned}\det(\boldsymbol{R}_{x'x'}) &= \det(\boldsymbol{E}_1^{\mathrm{T}}\boldsymbol{r}_{11}\boldsymbol{E}_1 + \boldsymbol{E}_2^{\mathrm{T}}\boldsymbol{r}_{22}\boldsymbol{E}_2) \geqslant \det(\boldsymbol{E}_1^{\mathrm{T}}\boldsymbol{r}_{11}\boldsymbol{E}_1) - \det(\boldsymbol{E}_2^{\mathrm{T}}\boldsymbol{r}_{22}\boldsymbol{E}_2) \\ &= \det(\boldsymbol{E}_1^{\mathrm{T}}\boldsymbol{r}_{11}\boldsymbol{E}_1)\end{aligned} \tag{9-61}$$

所以，有

$$\det(\boldsymbol{R}_{x'x'}) = \det(\boldsymbol{E}_1^{\mathrm{T}}\boldsymbol{r}_{11}\boldsymbol{E}_1) = \det(\boldsymbol{r}_{11})\det(\boldsymbol{E}_1^{\mathrm{T}}\boldsymbol{E}_1) \tag{9-62}$$

由相关的定义可知，

$$\det(\boldsymbol{E}_1^{\mathrm{T}}\boldsymbol{E}_1) = [(N-1)!]^2 > N\det(\boldsymbol{r}_{22}) \Rightarrow \det(\boldsymbol{R}_{x'x'}) \geqslant N\det(\boldsymbol{R}_{xx}) \tag{9-63}$$

结合上面各式可得，

$$\gamma(\boldsymbol{R}_{x'x'}) \leqslant \frac{\mathrm{tr}(\boldsymbol{E}^{\mathrm{T}}\boldsymbol{E})\,\mathrm{tr}(\boldsymbol{R}_{xx})}{N\det(\boldsymbol{R}_{xx})} = \frac{\mathrm{tr}(\boldsymbol{E}^{\mathrm{T}}\boldsymbol{E})}{N}\gamma(\boldsymbol{R}_{xx}) = \frac{2N^3+N}{3N^3}\gamma(\boldsymbol{R}_{xx}) \tag{9-64}$$

当 N 较大时，由式（9-64）可得以下关系：

$$\gamma(\boldsymbol{R}_{x'x'}) \leqslant \frac{2}{3}\gamma(\boldsymbol{R}_{xx}) < \gamma(\boldsymbol{R}_{xx}) \tag{9-65}$$

式（9-65）表明输入信号经过加权重叠处理后相关矩阵的特征值分散程度减小，因此提高了收敛速度。由式（9-52）又可看出由于步长减小，所以稳态误差也较小于 LMS 算法。上面对 ap^{1n} 的情况进行了分析，对于 ap^{1s} 结论仍然成立；对于 ap^{1d} 将无法保证式（9-65）成立，因此收敛速度比 LMS 低但稳态误差比无窗和单窗系统都低。另一方面，滤波器阶数的大小和正交基的选择对 W-O 算法的改善程度都有影响，这将在下面的实验中得以证实。

9.5　变换域加权重叠 LMS 算法（WO-TLMS）

变换域自适应滤波的概念是由 Dentino 等人于 1979 年首先提出的，其基本思想是把时域信号通过正交变换转变为变换域信号后采用自适应算法。其基本步骤是：①选择正交变换，把时域信号变换成变换域信号；②变换后的信号能量归一化；③采用某一自适应算法进行滤波。常用的正交变换有离散傅里叶变换、离散余弦变换、离散正弦变换、离散小波变换以及沃尔什-哈达玛等。

经过正交变换后的输入信号的自相关矩阵基本上具有稀疏带状结构，从而提高了收敛速度。但由于变换是基于信号分段进行的，因此在输出分段拼接处会出现混叠而失真，并在随后的自适应估计中产生较大的误差而影响算法的收敛性能。针对这种情况，本节提出了使用加权重叠滤波的思想以克服上述缺点，以改进传统变换域自适应算法的收敛性能。加权重叠变换域自适应滤波器可以用图 9-10所示。

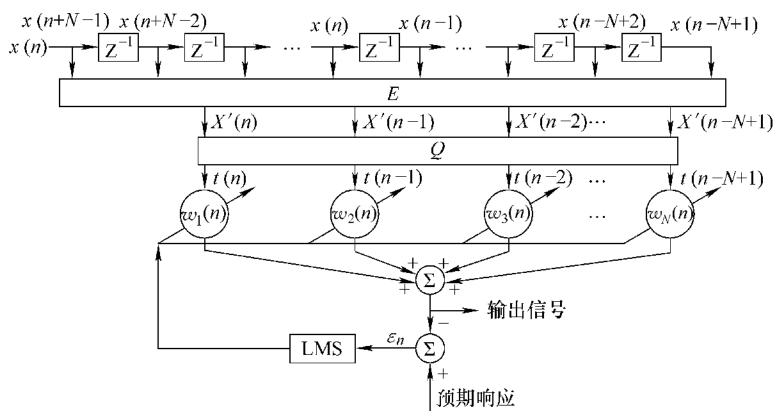


图 9-10 WO-TLMS 自适应滤波器信号流图

图 9-10 中 $\boldsymbol{E}$ 是全相位转移矩阵，$\boldsymbol{Q}$ 是 $N \times N$ 阶的正交矩阵，$\boldsymbol{t}(n)$ 是正交变换的输出。由图 9-10 可以看出，自适应过程可以分 4 步进行：第一，对输入加窗重叠；第二，正交变换；第三，求出输出和误差；第四，系数调整。以上过程可以表示如下：

$$\begin{cases} \boldsymbol{x}'(n) = \boldsymbol{E}^{\mathrm{T}}\boldsymbol{x}(n) \\ \boldsymbol{t}(n) = \boldsymbol{Q}\boldsymbol{x}'(n) \\ y_n = \boldsymbol{t}^{\mathrm{T}}(n)\boldsymbol{w}_n \\ \boldsymbol{\varepsilon}_n = d_n - y_n \\ \boldsymbol{w}_{n+1} = \boldsymbol{w}_n + 2\bar{\mu}\varepsilon_n\boldsymbol{t}(n) \end{cases} \tag{9-66}$$

式中，$\bar{\boldsymbol{\mu}} = \mathrm{diag}(\mu_0 \cdots \mu_{N-2} \quad \mu_{N-1})$，其中任一元素由下式求出：

$$\mu_i = \mu/\boldsymbol{P}_n(i) \quad i = 0,\ 1,\ \cdots,\ N-1 \tag{9-67}$$

式中，μ 是调节步长；$P_n(i)$ 是 n 时刻第 i 个频点输入功率的估值，可由下述递推公式求出：

$$\boldsymbol{P}_{n+1}(i) = (1-\alpha)\boldsymbol{P}_n(i) + \alpha\,|\boldsymbol{T}_n(i)|^2 \tag{9-68}$$

式中，$\boldsymbol{T}_n$ 是 $\boldsymbol{t}_n$ 的正交变换；α 是常数。由式(9-66) ~式(9-68) 可以得到：

$$\begin{cases} y_n = \boldsymbol{w}_n^{\mathrm{T}}[\boldsymbol{E}_{\mathrm{t}}\boldsymbol{x}(n)] \\ \boldsymbol{w}_{n+1} = \boldsymbol{w}_n + 2\bar{\mu}\varepsilon_n[\boldsymbol{E}_{\mathrm{t}}\boldsymbol{x}(n)] \end{cases} \tag{9-69}$$

式中，$\boldsymbol{E}_{\mathrm{t}} = \boldsymbol{Q}\boldsymbol{E}^{\mathrm{T}} = \boldsymbol{QOW}$。由上面的结果可以看出，带窗重叠变换域自适应的滤波器的变换矩阵等于 $\boldsymbol{E}_{\mathrm{t}}$。令 $\boldsymbol{E}_{\mathrm{t}}\boldsymbol{x}(n) = \boldsymbol{z}(n)$，采用同普通 LMS 自适应滤波器相类似的方法可得 WO-TLMS 自适应滤波器权矢量的维纳解 $\boldsymbol{w}_{\mathrm{opt}}$ 和相应的最小均方误差 $\boldsymbol{\varepsilon}_{\min}$ 分别为

$$
\begin{aligned}
\boldsymbol{w}_{\text{opt}} &= \boldsymbol{R}_{\text{zz}}^{-1}\boldsymbol{R}_{\text{zd}} = (\boldsymbol{E}_{\text{t}}\,\boldsymbol{R}_{\text{xx}}\boldsymbol{E}_{\text{t}}^{\text{T}})^{-1}\boldsymbol{E}_{\text{t}}\,\boldsymbol{R}_{\text{xd}} \\
&= (\boldsymbol{Q}\,\boldsymbol{E}^{\text{T}}\,\boldsymbol{R}_{\text{xx}}\boldsymbol{E}\,\boldsymbol{Q}^{\text{T}})^{-1}\boldsymbol{Q}\,\boldsymbol{E}^{\text{T}}\,\boldsymbol{R}_{\text{xd}} \\
&= \boldsymbol{Q}\,\boldsymbol{R}_{\text{x}'\text{x}'}^{-1}\boldsymbol{R}_{\text{x}'\text{d}} \\
&= \boldsymbol{Q}\boldsymbol{w}'_{\text{opt}}
\end{aligned}
\tag{9-70}
$$

$$
\begin{aligned}
\varepsilon_{\min}(n) &= E[d^2(n)] - \boldsymbol{R}_{\text{xd}}^{\text{T}}\boldsymbol{E}_{\text{t}}^{\text{T}}\boldsymbol{Q}\boldsymbol{w}'_{\text{opt}} \\
&= E[d^2(n)] - \boldsymbol{R}_{\text{x}'\text{d}}^{\text{T}}\boldsymbol{w}'_{\text{opt}} = \boldsymbol{\varepsilon}'^{WOT}_{\min}(n)
\end{aligned}
\tag{9-71}
$$

式中，$\boldsymbol{w}'_{\text{opt}}$和 $\varepsilon_{\min}$（n）分别是输入为 $\boldsymbol{x}'(n)$ 时正交变换 LMS 自适应滤波器权矢量维纳解与相应最小均方误差。WO-TLMS 中加权重叠矩阵对自适应算法的改善效果与 WO-LMS 一样，但采用不同的正交变换对系统性能的改善程度也会不同。

9.6　WO-LMS 在自适应辨识中的应用

自适应滤波算法被广泛地应用于系统辨识、回波抵消、自适应谱线增强、自适应信道均衡、语音线性预测、图像自适应滤波等诸多领域。图 9-11 为一个自适应辨识原理图。图中 $\boldsymbol{x}(m)$ 表示系统在 m 时刻的输入信号矢量，M 是自适应滤波器权数目，$n(m)$ 表示 m 时刻的干扰噪声，$\boldsymbol{w}^*$ 表示被辨识系统的权矢量，$\boldsymbol{w}(m)$ 是自适应滤波器在 m 时刻的权矢量，$d(m)$ 是 m 时刻的期望输出值，$e(m)=d(m)-\boldsymbol{w}^{\text{T}}(m)\boldsymbol{x}(m)$是误差信号，$y(m)$ 是 m 时刻的输出。由图 9-11 可知：

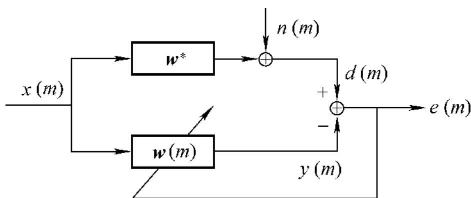


图 9-11　自适应辨识原理图

$$
\begin{aligned}
&\begin{cases} d(m) = \boldsymbol{w}^{*\text{T}}\boldsymbol{x}(m) + n(m) \\ y(m) = \boldsymbol{w}^{\text{T}}(m)\boldsymbol{x}(m) \end{cases} \\
\Rightarrow e(m) &= d(m) - y(m) \\
&= [\boldsymbol{w}^* - \boldsymbol{w}(m)]^{\text{T}}\boldsymbol{x}(m) + n(m) \\
&= \boldsymbol{v}^{\text{T}}(m)\boldsymbol{x}(m) + n(m)
\end{aligned}
\tag{9-72}
$$

因为 $\boldsymbol{x}$（m）与 n（m）不相关，所以

$$
\xi(m) = E[e^2(m)] = \boldsymbol{v}^{T}\boldsymbol{R}_{\text{xx}}\boldsymbol{v} + \boldsymbol{R}_{\text{nn}}
\tag{9-73}
$$

可以看出，只有 $\boldsymbol{v}=\boldsymbol{0}$ 即 $\boldsymbol{w}(m)=\boldsymbol{w}^*$ 时，均方误差才取得最小值。所以，LMS 算法收敛时，自适应滤波器的滤波系数等于被辨识的权矢量。基于 WO 算法的自适应辨识系统收敛时的精度比 LMS 提高很多，模拟实验条件如下：

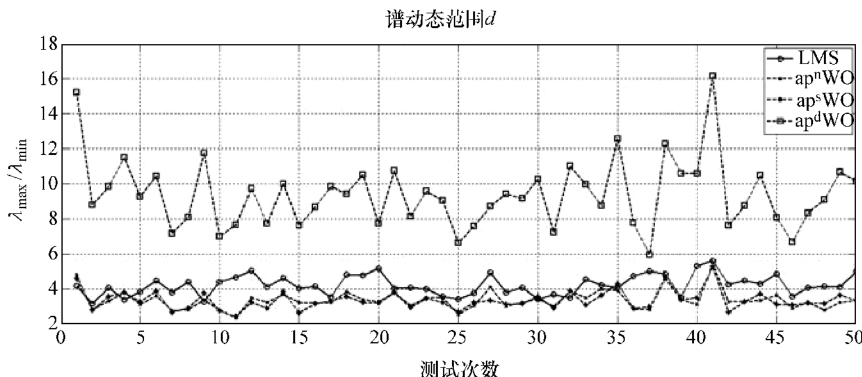
1）自适应滤波器阶数 $N=8$。

2）未知系统的 FIR 系数为：$\boldsymbol{h}^* = [0.9501\ 0.2311\ 0.6068\ 0.4860\ 0.8913\ 0.7621\ 0.4565\ 0.0185]^T$。

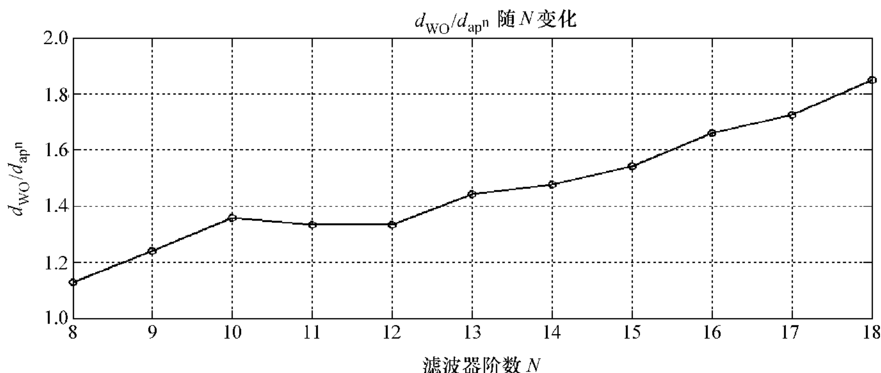
3）参考输入信号 $\boldsymbol{x}(n)$ 是高斯白噪声平稳序列，$E\{\boldsymbol{x}(n)\}=0$，$E\{\boldsymbol{x}^2(n)\}=1$。

4）$\boldsymbol{v}(n)$ 是与 $\boldsymbol{x}(n)$ 不相关的高斯白噪声，$E\{\boldsymbol{v}(n)\}=0, E\{\boldsymbol{v}^2(n)\}=0.01$。

为了说明 WO-LMS 算法的收敛性能，实验中首先对 WO-LMS 和 LMS 算法输入信号的特征值分散度进行了比较，结果如图 9-12 所示。图 9-12a 是 8 阶自适应滤波器的 WO-LMS 和 LMS 算法 50 次实验的输入自相关分散程度曲线。从图中可见 WO-LMS 算法比 LMS 算法有较高的收敛性能，且改善程度随滤波器阶数增加而增大。图 9-12b 是 3～18 阶滤波器 LMS 算法与 WO-LMS 算法分别进行 50 次独立实验统计平均谱分散程度比曲线。



a) WO与LMS算法谱动态范围比较



b) ap|n WO与LMS算法谱动态范围比随阶数N的变化

图 9-12　WO-LMS 和 LMS 算法输入自相关矩阵的分散比较

下面将给出在不同步长时，WO-LMS 算法响应收敛曲线和 LMS 算法收敛曲线的比较。曲线横坐标为迭代次数，纵坐标以系数误差分贝 10log10［$\|\boldsymbol{h}-\boldsymbol{h}^{*}\|$］作为算法收敛的性能指标（独立进行 50 次实验，每次实验 1000 个样本点），如图 9-13 所示。

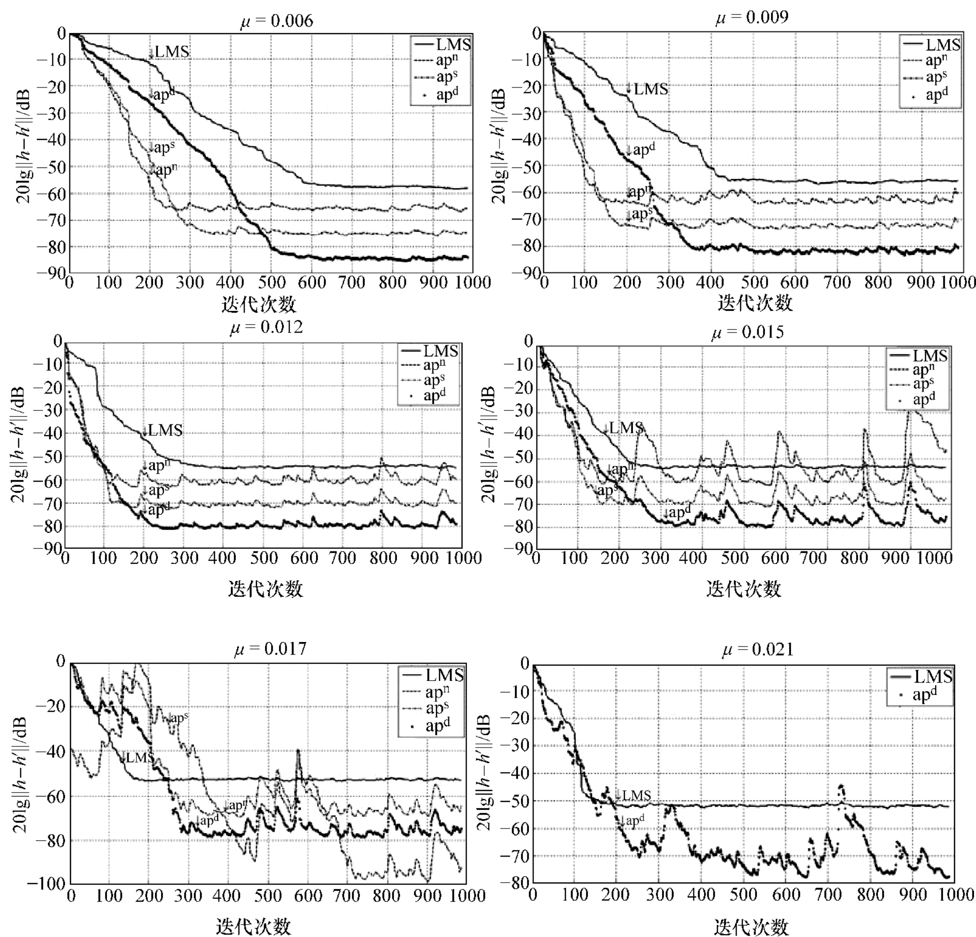


图 9-13　不同 μ 值 WO-LMS 算法和 LMS 算法系数收敛曲线对比

从图 9-13 实验结果可以看出，在步长较小时 WO-LMS 比 LMS 算法有较高的收敛速度和较低的稳态失调，其中 ap^{n} 和 ap^{s} 收敛速度大于 ap^{d}，但 ap^{d} 具有比 ap^{ns} 低得多的稳态失调且步长越小这种优势越明显；随着步长增大，使得 ap^{nsd} 稳态时失调摆动增大但仍明显低于 LMS 算法；$\mu=0.021$ 时 ap^{ns} 算法已经发散而 ap^{d} 算法有相当的收敛速度而稳态失调低于 LMS 算法近 20dB。实现上面实验 MATLAB 代码如下：

```
function [wc, wwo] =woAR

clc, clf, close all;
u=0.021; M=1000;
wopt=[0.9501; 0.2311; 0.6068; 0.4860; 0.8913; 0.7621; 0.4565; 0.0185];
N=length (wopt);
signal=randn (1, M); noise=0.01* randn (1, M);

wins (:, 1) =conv (ones (1, N), ones (1, N));
wins (:, 2) =conv (ones (1, N), [1: N/2, N/2: -1: 1]);
wins (:, 3) =conv ( [1: N/2, N/2: -1: 1], [1: N/2, N/2: -1: 1]);

% [dx dxp] =dcomp1 (signal, N, wins);
% decomp2 (signal, N);
for id=0: 3
   erlg (:, id+1) =apAR (N, u, signal, noise, wopt, id, wins);
   end
   lnm= {'\downarrowLMS', '\downarrowap^n', '\downarrowap^s', '\downar-
rowap^d'};
len=size (erlg, 1); lns= {'k-', 'k--', 'k-.', 'k.'}; hold  on;

for i=1: 4
   plot (erlg (:, i), lns {i}, 'LineWidth', 2);
   text (fix (len/5), erlg (fix (len/5), i) +3, lnm {i}, 'FontSize', 16);
end
legend ('LMS', 'ap^n', 'ap^s', 'ap^d');
grid on; set (gca, 'FontSize', 16);
xlabel ('iteration times'); ylabel ('20lg||h-h^*|| (dB) ');
title (strcat ('\it\mu\rm=', num2str (u)));

function decomp2 (signal, N)
forN=8: 18
   wins=conv (boxcar (N), boxcar (N)); wins=wins/max (wins);
   [dx dxp] =dcomp1 (signal, N, wins);
   dx2xp (N-7) =mean (dx (:, 1)) /mean (dxp (:, 1));
end
figure, plot ( [8: 18], dx2xp, 'ko-', 'LineWidth', 2); grid on; set (gca, '
FontSize', 16);
xlabel ('Filter Order \itN');
ylabel ('\it\bfd\rm_ {WO} /\it\bfd\rm_ {ap^n} ');
title ('\it\bfd\rm_ {WO} /\it\bfd\rm_ {ap^n} to \itN');
```

```
function [dx dxp] =dcomp1 (signal, N, wins)
sig= [fliplr (signal (1: N -1)), signal, fliplr (signal (end -N + 2: end))];
M=length (signal); n_ m=fix (M/N); wlen=size (wins, 2);
for wi =1: wlen
    for mi =1: n_ m
        x=sig ( (mi -1) * N + 1: (mi + 1) * N -1);
        xpie=sig ( (mi -1) * N + 1: (mi + 1) * N -1);
        wxpie=wins (:, wi) '.* xpie;
        woxpie= [wxpie (1: N -1) + wxpie (N + 1: 2* N -1) wxpie (N)];
        xeig (:,:, mi) =x'* x; xpieeig (:,:, mi) =woxpie'* woxpie;
    end
    M=100; ei=fix (n_ m/M);
    for mi =1: ei
        xe (:,:, mi) =mean (xeig (:,:, (mi -1) * M + 1: mi* M), 3);
        xpe (:,:, mi) =mean (xpieeig (:,:, (mi -1) * M + 1: mi* M), 3);
        dx (mi, wi) =max (eig (xe (:,:, mi))) /min (eig (xe (:,:, mi)));
        dxp (mi, wi) =max (eig (xpe (:,:, mi))) /min (eig (xpe (:,:, mi)));
    end
end

h (1) =plot (dx (:, 1), 'ko -'); hold on;
h (2) =plot (dxp (:, 1), 'kx - -'); h (3) =plot (dxp (:, 2), 'k* - -');
h (4) =plot (dxp (:, 3), 'ks - -'); set (gca, 'FontSize', 16); set (h, 'Line-
Width', 2);
hold off; legend ('LMS', 'ap^n WO', 'ap^s WO', 'ap^d WO');
xlabel (' \itn \rmth test');
ylabel (' \it \lambda \rm_ {max} / \it \lambda \rm_ {min} ');
title ('spectrum dispersity \it \bfd'); grid on;

function erlg=apAR (N, u, signal, noise, wopt, id, wins)
M=50; sam=length (signal);

for m=1: M
    w=zeros (N, 1); noise=0.01* randn (1, sam);
    for j =1: sam -2* N + 2
      if id ~ =0
          x=signal (j: j + 2* N -2); wox=x. * wins (:, id) ';
          x= [wox (N), wox (1: N -1) + wox (N + 1: end)];
     else
          x=signal (j: j + N -1);
          end
```

```
        err (j, m) = (wopt - w) '* (wopt - w);
        e (j) =x* wopt + noise (j + N -1) -x* w;
        w =w + 2* u/N^id* e (j) * x.';
    end
end
er =sum (err, 2) /M; erlg =10* log10 (er); erlg =erlg - max (erlg);
```

对于 WO – TLMS 的测试，选用的 3 种正交变换为 DFT 和 DCT ，实验条件如下：

1）自适应滤波器阶数 $N=8$。

2）未知系统的 FIR 系数为：$\boldsymbol{h}^* = [0.9501\ 0.2311\ 0.6068\ 0.4860\ 0.8913\ 0.7621\ 0.4565\ 0.0185]^{\mathrm{T}}$。

3）参考输入信号 $\boldsymbol{x}(n)$ 是高斯白噪声平稳序列，$E\{\boldsymbol{x}(n)\}=0, E\{\boldsymbol{x}^2(n)\}=1$。

4）$\boldsymbol{v}(n)$ 是与 $\boldsymbol{x}(n)$ 不相关的高斯白噪声，$E\{\boldsymbol{v}(n)\}=0, E\{\boldsymbol{v}^2(n)\}=0.01$。

首先对 ap^{1nsd} 系统在分别采用 FFT 和 DCT 两种正交变换情况下对输入信号的谱动态范围进行比较。分别进行 50 次独立的实验，每次实验样本点为 1000 点。实验结果如图 9-14 所示（曲线纵坐标表示 WO-TLMS 算法对 TLMS 算法中输入信号的谱动态范围之比，横坐标是实验次数）。

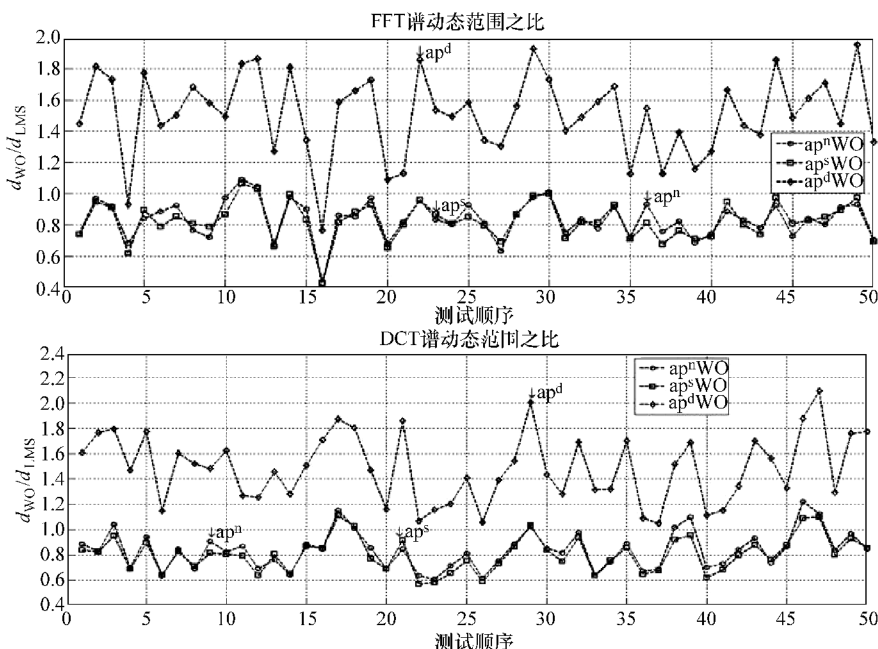


图 9-14 FFT/DCT ap^{1nsd} WO-TLMS 与 TLMS 谱动态范围比

由图 9-14 可以看出，两种正交变换下 ap^{ns} WO-TLMS 算法中对输入信号的处理都使得谱动态范围小于传统 TLMS 算法。ap^{d} 使得收敛速度变慢，分散度最大偏离 TLMS 算法的 2 倍左右；DCT 对收敛性能的改善同 DFT 基本相当。

下面给出 ap^{nsd} WO-TLMS 算法收敛曲线和 TLMS 算法的收敛曲线对比。分别进行 50 次独立实验，每次实验样本点为 1000，然后求其统计平均得出学习曲线，如图 9-15 所示。曲线的横坐标表示迭代次数，纵坐标以系数分贝误差。参数选取步长 $\mu = 0.025$，功率归一化因子 $\alpha = 0.8$。

由图 9-14 和图 9-15 可以看出，两种正交变换下 ap^{ns} WO-TLMS 算法的收敛速度大于 TLMS 而且稳态误差也小于 TLMS 算法；ap^{d} WO-TLMS 算法的收敛速度低于 TLMS，但它可以获得比 TLMS 低得多的稳态失调，而且两种正交变换中双窗算法的稳态误差都低于无窗和单窗算法。FFT 得到的曲线收敛精度较高但收敛速度慢，DCT 具有较快的收敛速度但稳态误差明显低于 DFT。表 9-1 是各算法达到稳态后的权系数估值（精确到小数点后 4 位）。

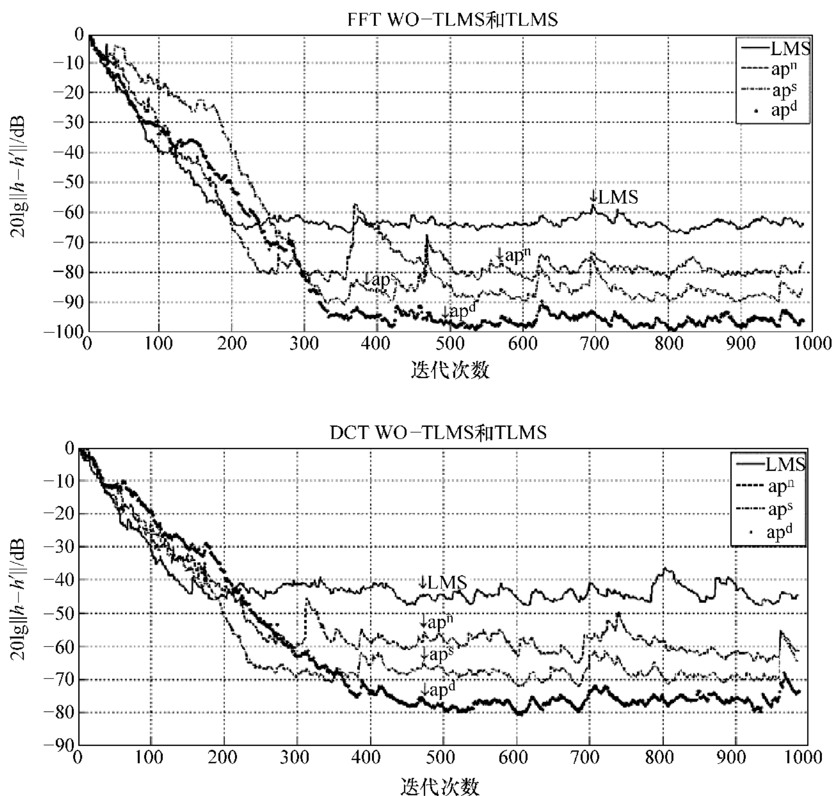


图 9-15　FFT/DCT ap^{nsd} WO-TLMS 与 TLMS 学习曲线

表 9-1 FFT/DCT ap^{1nsd} WO-TLMS 算法与 TLMS 算法稳态估值

分类		h_0	h_1	h_2	h_3	h_4	h_5	h_6	h_7
DFT	LMS	0.9500	0.2309	0.6064	0.4856	0.8916	0.7622	0.4565	0.0188
	ap^n	0.9499	0.2311	0.6067	0.4859	0.8912	0.7623	0.4563	0.0188
	ap^s	0.9501	0.2311	0.6068	0.4860	0.8913	0.7622	0.4565	0.0185
	ap^d	0.9501	0.2311	0.6068	0.4860	0.8913	0.7621	0.4565	0.0185
DCT	LMS	0.9548	0.2325	0.6114	0.4863	0.8914	0.7568	0.4562	0.0112
	ap^n	0.9496	0.2306	0.6061	0.4838	0.8908	0.7613	0.4553	0.0176
	ap^s	0.9504	0.2311	0.6072	0.4864	0.8917	0.7626	0.4571	0.0188
	ap^d	0.9501	0.2311	0.6067	0.4860	0.8915	0.7621	0.4566	0.0185

由上面时域和变换域 WO-TLMS 在自适应辨识中的应用可以看出，ap^{1nsd}系统对应的 WO 算法都明显提高了收敛精度。其中 ap^{1d}具有最好的估值精度，但收敛速度低于 ap^{1ns}；ap^{1n}的收敛速度最快，但精度低于 ap^{1sd}。由于在自适应辨识中，主要关注的是收敛后的稳态误差大小，因此，ap^{1d} WO-TLMS 算法是较为理想的改善方案。

实现上面实验的 MATLAB 代码（同时实现时域和频域）如下：

```
function [wc, wwo] =wotAR

clc, clf, close all;
u =0.026; M =1000;
wopt = [0.9501; 0.2311; 0.6068; 0.4860; 0.8913; 0.7621; 0.4565; 0.0185];
N =length (wopt);
signal =randn (1, M); noise =0.01* randn (1, M);

wins (:, 1) =conv (ones (1, N), ones (1, N));
wins (:, 2) =conv (ones (1, N), [1: N/2, N/2: -1: 1]);
wins (:, 3) =conv ( [1: N/2, N/2: -1: 1], [1: N/2, N/2: -1: 1]);

% dcomp1 (signal, N, wins, 2);% decomp2 (signal, N);
start =4; stop =7;% 0: LMS, 1 ~3: apnsd, 4 ~7: aptnsd;
for id =start: stop
   erlg (:, id - start + 1) =apAR (N, u, signal, noise, wopt, id, wins);
end
lnm = {'\downarrowLMS', '\downarrowap^n', '\downarrowap^s', '\downarrowap
^d'}; len =size (erlg, 1); lns = {'k -', 'k - -', 'k -.', 'k.'}; hold on;

for i =1: stop - start + 1
   plot (erlg (:, i), lns {i}, 'LineWidth', 2);
```

```
    text (fix (len/5), erlg (fix (len/5), i) + 3, lnm { i }, 'FontSize', 16);
end
legend ('LMS', 'ap^n', 'ap^s', 'ap^d'); grid on; set (gca, 'FontSize', 16);
xlabel ('iteration times'); ylabel ('20lg | |h - h^* | | (dB) ');
if id < 3
    title (strcat ('\it\mu\rm=', num2str (u)));
else
    title ('FFT WO - TLMS and TLMS');
end

function decomp2 (signal, N)
for N =8: 18
    wins =conv (boxcar (N), boxcar (N)); wins =wins/max (wins);
    [dx dxp] =dcomp1 (signal, N, wins, 0);
    dx2xp (N -7) =mean (dx (:, 1)) /mean (dxp (:, 1));
end
figure, plot ([8: 18], dx2xp, 'ko -', 'LineWidth', 2);
grid on; set (gca, 'FontSize', 16);
xlabel ('Filter Order \itN');
ylabel ('\it\bfd\rm_ {WO} /\it\bfd\rm_ {ap^n} ');
title ('\it\bfd\rm_ {WO} /\it\bfd\rm_ {ap^n} to \itN');

function [dx dxp] =dcomp1 (signal, N, wins, id)
sig= [fliplr (signal (1: N -1)), signal, fliplr (signal (end -N + 2: end))];
M =length (signal); n_ m =fix (M/N); wlen =size (wins, 2);
switch id
    case 0
        xs ='x =sig ( (mi -1) * N + 1: (mi + 1) * N -1); ';
        xps ='woxpie = [wxpie (N) wxpie (1: N -1) + wxpie (N + 1: 2* N -1)]; ';
    case 1
        xs ='x =fft (sig ( (mi -1) * N + 1: (mi + 1) * N -1)); ';
        xps ='woxpie =fft ([wxpie (N) wxpie (1: N -1) +wxpie (N +1: 2* N -1)]);
';
    case 2
        xs ='x =dct (sig ( (mi -1) * N + 1: (mi + 1) * N -1)); ';
        xps ='woxpie =dct ([wxpie (N) wxpie (1: N -1) + wxpie (N + 1: 2 * N -
1)]); ';
    otherwise
        error ('Bad Parameter');
end
```

```
for wi =1: wlen
    for mi =1: n_ m
        eval (xs); xpie=sig ( (mi -1) * N + 1: (mi + 1) * N -1);
        wxpie=wins (:, wi) '.* xpie; eval (xps);
        xeig (:,:, mi) =x'* x; xpieeig (:,:, mi) =woxpie'* woxpie;
    end
    M =100; ei =fix (n_ m/M);
    for mi =1: ei
        xe (:,:, mi) =mean (xeig (:,:, (mi -1) * M + 1: mi* M), 3);
        xpe (:,:, mi) =mean (xpieeig (:,:, (mi -1) * M + 1: mi* M), 3);
        dx (mi, wi) =max (eig (xe (:,:, mi))) /min (eig (xe (:,:, mi)));
        dxp (mi, wi) =max (eig (xpe (:,:, mi))) /min (eig (xpe (:,:, mi)));
    end
end
if id = =0
    h (1) =plot (dx (:, 1), 'ko -'); hold on;
    h (2) =plot (dxp (:, 1), 'kx - -'); h (3) =plot (dxp (:, 2), 'k* - -');
    h (4) =plot (dxp (:, 3), 'ks - -');
    set (gca, 'FontSize', 16); set (h, 'LineWidth', 2);
    hold off; legend ('LMS', 'ap^n WO', 'ap^s WO', 'ap^d WO');
    xlabel ('\itn \ rmth test');
    ylabel ('\it\lambda\rm_ {max} /\it\lambda\rm_ {min} ');
    title ('spectrum dispersity \it\bfd'); grid on;
else
    lns= { 'ko - -', 'ks - -', 'kd - -' }; hold on;
    lnm= { '\downarrowap^n', '\downarrowap^s', '\downarrowap^d' };
    for i =1: wlen
        dr=dxp (:, i) ./dx (:, 1);
        h (i) =plot (dr, lns { i }); xcor=fix (ei/2);
        text (xcor, dr (xcor) * 1.1, lnm { i }, 'FontSize', 16);
    end

    set (gca, 'FontSize', 16); set (h, 'LineWidth', 2);
    hold off; legend ('ap^n WO', 'ap^s WO', 'ap^d WO');
    xlabel (' \itn\rmth test'); ylabel ('\itd\rm_ {WO} /d_ {LMS} ');
    title ('spectrum dispersity ratio'); grid on;
end

function erlg=apAR (N, u, signal, noise, wopt, id, wins)
M =50; sam=length (signal); b=0.8;
sfun1 =' x=signal (j: j + 2* N -2); wox=x.* wins (:, id) ''; ';
```

```
sfun2 = 'if j = =1; p = abs (x) .^2; end ';
sfun3 = ' w = w + 2* u/N^id* e (j) * x.''; ';
switch id
    case 0
        sfun1 = ''; sfun2 = '';
        xin = 'signal (j: j + N -1); ';
    case { 1 2 3 }
        xin = ' [wox (N), wox (1: N -1) + wox (N + 1: end)]; ';
        sfun2 = '';
        case 4
        sfun1 = ''; xin = 'dct (signal (j: j + N -1)); ';
        wopt = dct (wopt, N);
        sfun3 = 'p = (1 -b) * p + b* abs (x) .^ 2; w = w +2 * u* e (j) * (x. /p) '
'; ';
    case { 5 6 7 }
        id = id -4;
        xin = ' dct ( [wox (N), wox (1: N -1) + wox (N + 1: end)]); ';
        wopt = dct (wopt, N);
        sfun3 = 'p = (1 -b) * p + b* abs (x) .^ 2; w = w +2* u* e (j) * (x. /p) '';
';
end

for m =1: M
    w = zeros (N, 1); noise =0.01* randn (1, sam);
    for j =1: sam -2* N + 2
      eval (sfun1); x = eval (xin); eval (sfun2);
      err (j, m) = real ( (wopt - w) '* (wopt - w));
      e (j) = x* wopt + noise (j + N -1) - x* w;
      eval (sfun3);
    end
end
er = sum (err, 2) /M; erlg =10* log10 (er); erlg = erlg - max (erlg);
idct (w)
```

9.7　WO-LMS 在自适应除噪中的应用

自适应除噪（ANC）系统的原理图如图 9-16 所示。

在 ANC 中通常假定参考通道噪声 $n_r(n)$ 是由主通道噪声 $n_m(n)$ 经过一个传递函数为 $H(z)$ 的系统的输出，自适应除噪的本质就是通过自适应算法估计出 $H(z)$ 的逆，用 $n_r(n)$ 逼近 $d(n) = s(n) + n_m(n)$，而有用信号与噪声不相关。

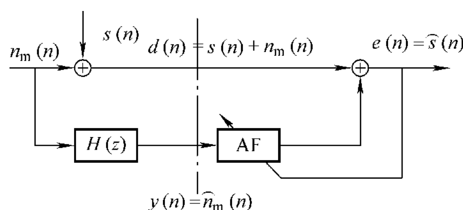


图 9-16　ANC 原理图

实际上自适应算法是用 $n_r(n)$ 去逼近 $n_m(n)$，得到 $n_m(n)$ 的估计值 $\hat{n}_m(n)$，再与主通道 $d(n)$ 相减以达到最大限度地消除主通道噪声 $n_m(n)$ 以获得有用信号 $s(n)$。自适应滤波器的权系数是由逼近误差为 $e(n)$ 通过 LMS 准则进行反馈调整。由图 9-16 可知系统的输出 $e(n)$ 为：

$$\begin{aligned} &e(n)=s(n)+n_m(n)-\hat{n}_m(n) \\ \Rightarrow &e^2(n)=s^2(n)+[n_m(n)-\hat{n}_m(n)]^2+2s(n)[n_m(n)-\hat{n}_m(n)] \end{aligned} \tag{9-74}$$

两边取数学期望，由于 $s(n)$ 与 $n_m(n)$ 和 $\hat{n}_m(n)$ 均不相关，所以有

$$E[e^2(n)]=E[s^2(n)]+E[[n_m(n)-\hat{n}_m(n)]^2] \tag{9-75}$$

信号功率 $E[s^2(n)]$ 与自适应滤波器的调节无关，故自适应调节使 $E[e^2(n)]$ 最小就是使 $E[\langle n_m(n)-\hat{n}_m(n)\rangle^2]$ 最小。由式（9-74）可知，$n_m(n)-\hat{n}_m(n)=e(n)-s(n)$，当 $E[\langle n_m(n)-\hat{n}_m(n)\rangle^2]$ 最小时 $E[\langle e(n)-s(n)\rangle^2]$ 也最小，即自适应噪声抵消系统的输出信号 $e(n)$ 与有用信号 $s(n)$ 的均方误差最小。可以证明，自适应滤波器能完成上述任务的必要条件为：参考输入信号 $n_r(n)$ 必须与被抵消的信号 $n_m(n)$ 相关。在图 9-16 系统中，$n_r(n)$ 是 $n_m(n)$ 通过一线性系统的输出，所以收敛条件满足。

分别用传统 LMS 算法和 ap^{1nsd} WO-LMS 算法进行自适应除噪，模拟条件如下：

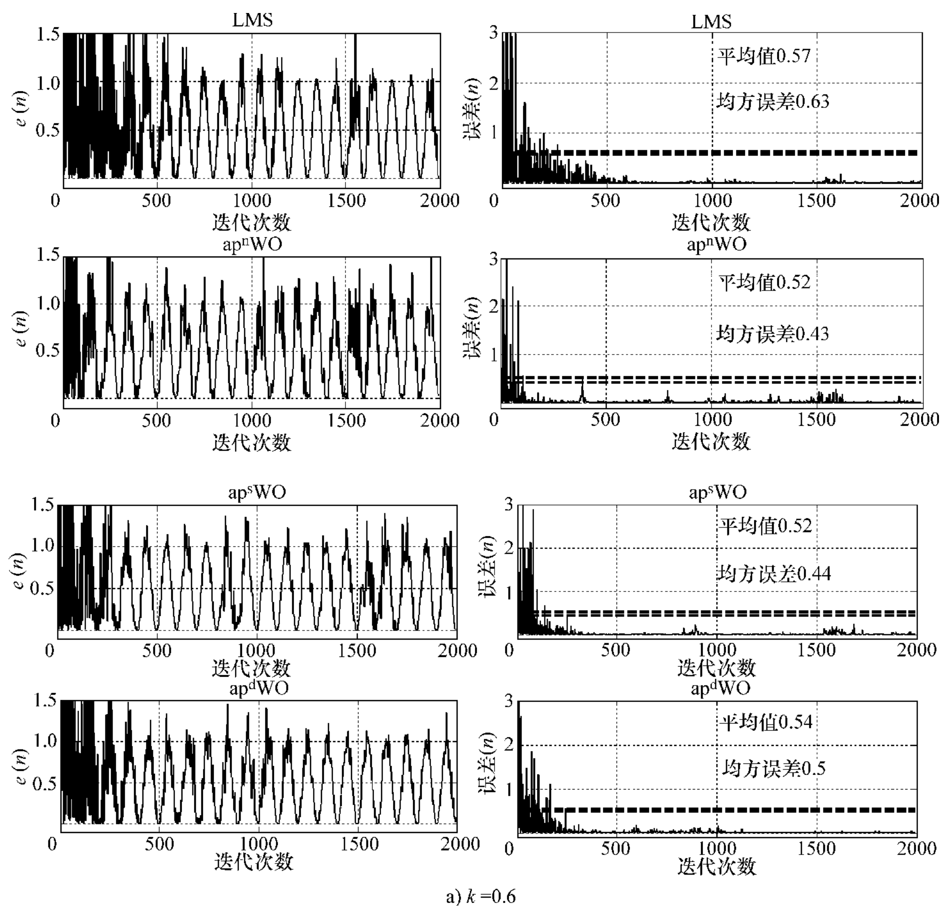
1）滤波器阶数 $N=4$。

2）信号 $s(n)=\sin(2n\pi/200)$。

3）主通道噪声 $n_m(n)$ 是均值为 0，方差为 1 的高斯白噪声。

4）参考噪 $n_r(n)$ 经过传递函数 $H(z)=k+(1-k)z^{-1}(0\leqslant k\leqslant 1)$ 得到，k 是参考噪声与主通道噪声的相关程度系数。

实验进行 2000 次迭代，收敛步长 $\mu=0.002$。实验结果如图 9-17 所示（横坐标表示迭代次数，纵坐标为误差幅值）。



a) $k=0.6$

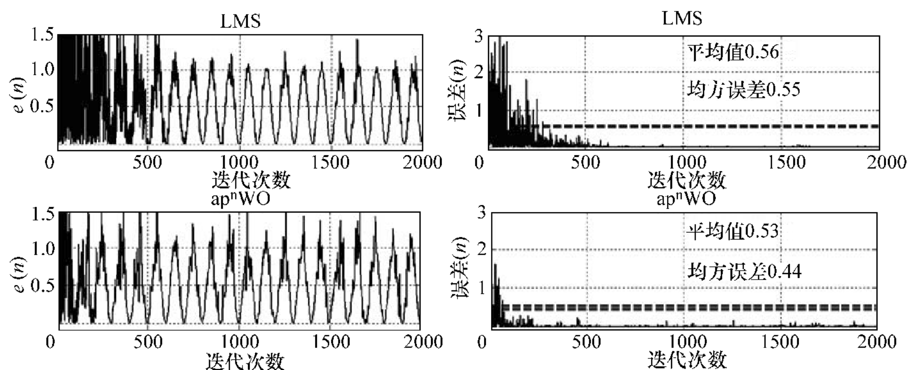
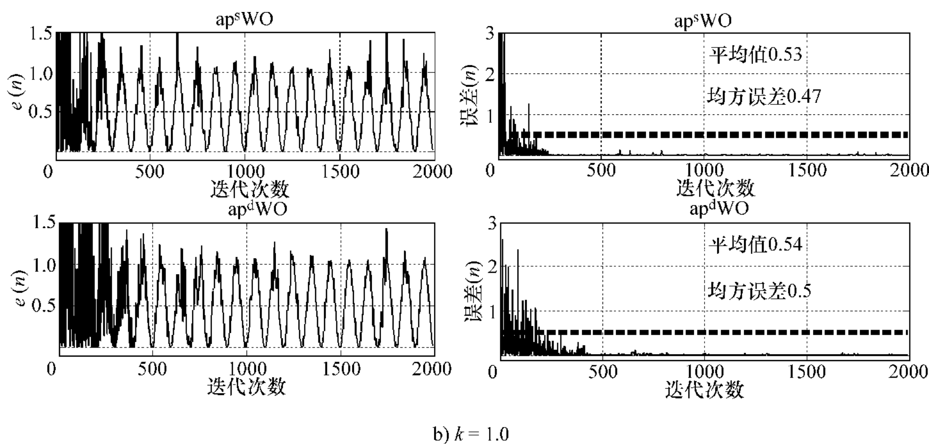


图 9-17　$k=0.6$ 和 $k=1.0$ 时 WO 和 LMS 除噪学习曲线



b) $k=1.0$

图 9-17 $k=0.6$ 和 $k=1.0$ 时 WO 和 LMS 除噪学习曲线（续）

图 9-17 给出了相关度为 60%（图 9-17a）和 100%（图 9-17b）时的除噪曲线。从图中可以看出，ap^{1ns} WO-LMS 的收敛速度明显大于 LMS，ap^{1d}收敛速度低于 ap^{1ns}，但稳态误差明显好于 ap^{1ns}和 LMS；当相关性由 60% 变为 100%，收敛速度和稳态误差均得到改善。然而，WO-LMS 算法的优点随相关的减弱而变的越来越不明显。在收敛速度基本相同的条件下，WO-LMS3 种算法得到的稳态误差随相关性的减小而递减。图 9-18 给出了各种算法的变化规律。

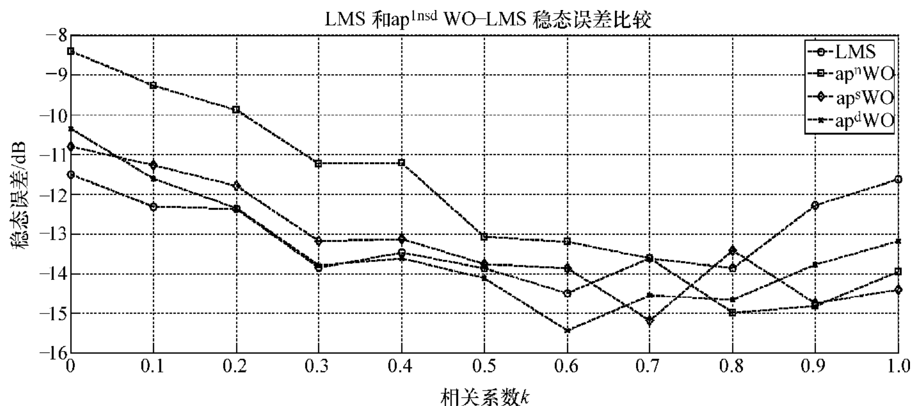


图 9-18 稳态均方误差关于相关性变化曲线

由图 9-18 可知，k 小于 0.3 时，LMS 具有最小的稳态误差；k 大于 0.3 后，ap^{1d}的稳态误差小于传统 LMS 算法；相关性大于 0.7 时，ap^{1nsd} WO-LMS 算法的

稳态误差都低于 LMS；ap^{1d}特性好于 ap^{1nsd}，ap^{1n}的性能最差。上面实验的 MATLAB 代码如下：

```
function woDN (N, sam)
if nargin = =0
   N =4; sam =2000; clc, clf, close all;
   end
   u =0.002; k =0.6;
   for k =0: 0.1: 1
      for id =0: 3
          [e err] =dnoise (N, sam, u, k, id);
         eres (id + 1,:) =e; errres (id + 1,:) =err;
      end
      kerr (:, uint8 (k/0.1 + 1)) =sum (errres, 2) /size (errres, 2);
end
tn = { 'LMS', 'ap^n WO', 'ap^s WO', 'ap^d WO' };
if 1
   mt = { 'o', 's', 'd', 'x' }; hold on;
   for i =1: 4
      h (i) =plot ([0: 0.1: 1], 10* log10 (kerr (i,:)), 'k - -', 'Marker', mt
{i});
   end
   set (h, 'MarkerSize', 8, 'LineWidth', 2);
   set (gca, 'FontSize', 16); grid on; legend (tn);
   xlabel ('correlation coeff. \itk'); ylabel ('stable err (dB) ');
   title ('LMS and ap^ {1nsd} WO - LMS err comparison');
else
   ewh =size (e);
   for i =1: 4
      if mod (i, 2) = =1
         figure, N =mod (i, 2);
      else
         n =n + 1;
      end
      subplot (2, 2, (n -1) * 2 + 1), plot (eres (i,:), 'k', 'LineWidth', 2);
      grid on; set (gca, 'FontSize', 16); set (gca, 'YLim', [ -0.1 1.5]);
      xlabel (' \itn \rmth iterations'); ylabel (' \ite \rm ( \itn \rm) ');
      title (tn {i});
      subplot (2, 2, (n -1) * 2 +2), plot (errres (i,:), 'k', 'LineWidth',
2);
      grid on; set (gca, 'FontSize', 16); set (gca, 'YLim', [ -0.1 3]);
      xlabel (' \itn \rmth iterations'); ylabel (' error \rm ( \itn \rm) ');
```

```
        title (tn {i});
        meres = fix (mean (eres (i,:)) * 100) /100;
        stderes = fix (std (eres (i,:)) * 100) /100;
        hold on, plot (ones (1, ewh (2)) * meres, 'k--', 'LineWidth', 3);
        plot (ones (1, ewh (2)) * stderes, 'k--', 'LineWidth', 3);
        text (ewh (2) /2, meres + 2, strcat ('mean value', num2str (meres)),
        'FontSize', 16);
        text (ewh (2) /2, stderes + 1, strcat ('std error',
        num2str (stderes)),  'FontSize', 16);
    end
end

function [e err] =dnoise (N, sam, u, k, id)
signal = sin (2* pi/200* [0: sam -1]); noise = randn (1, sam + 1);
refn =  k* noise (2: sam + 1) + (1 - k) * noise (1: sam);
wc = zeros (N, 1); ww = wc; input = signal + [noise (1) refn (1: sam -1)];
wins (:, 1) = conv (ones (1, N), ones (1, N));
wins (:, 2) = conv (ones (1, N), [1: N/2, N/2: -1: 1]);
wins (:, 3) = conv ( [1: N/2, N/2: -1: 1], [1: N/2, N/2: -1: 1]);

for j = N: sam - N + 1
    if id = = 0
        x = noise (j - N + 1: j); y = x* wc;
        e1 (j - N + 1) = input (j) - y;
        wc = wc + 2* u* e1 (j - N + 1) * x.';
        err1 (j - N + 1) = e1 (j - N + 1) - signal (j - N + 1);
    else
        xi = noise (j - N + 1: j + N -1) .* wins (:, id) ';
        x = [xi (1: N -1) + xi (N + 1: 2* N -1) xi (N)];
        yw = x* ww; e1 (j - N + 1) = input (j) - yw;
        ww = ww + 2* u* e1 (j - N + 1) /N^id* x.';
        err1 (j - N + 1) = e1 (j - N + 1) - signal (j - N + 1);
    end
end
e = e1.* e1; err = err1.* err1;
```

9.8 小结

本章首先介绍了基于 LMS 准则的自适应滤波器结构和算法，指出了收敛速度和稳态误差是一对矛盾体，并提出了改善收敛性能的 3 种方案；然后，将 ap

重叠加权思想与 LMS 算法相结合，提出时域和变换域重叠加权的 LMS 算法（WO-LMS、WO- TLMS），从理论上证明了 WO 算法比传统 LMS 算法提高了收敛速度与减小了稳态误差；最后，进行了基于 WO 算法的自适应辨识和自适应除噪实验，并对结果做了较详尽分析，论证 WO 算法的可行性和正确性。

对于二维信号的自适应处理，可以按照分析一维信号的方式来从行列两方向来进行，也可以直接使用模板并设定判定准则来建立自适应过程。对于三维视频的某些应用（如帧预测，数据预处理以降低相关性等）可以考虑采用 WO-LMS 算法。如何面向具体应用建立以某关注量为参数的评价函数而导入 WO 算法是值得研究的内容。

第 10 章 全相位处理的Simulink仿真

10.1 MATLAB 简介

前面 9 章中从理论上对 ap 信号处理方法及应用进行了分析和总结，并基于 MATLAB 工具设计实验验证结论的正确性。本章将借助 Simulink 仿真软件实现 ap 模型，并对信号处理过程进行仿真验证。

MATLAB 是由美国 Math Works 公司推出的用于数值计算和图形处理的科学系统环境。MATLAB 是英文 MATrix LABoratory（矩阵实验室）的缩写，它集中了日常数学处理中各种功能，包括高效的数值计算、矩阵计算、信号处理、图形绘制、输入输出和文件管理等各项操作。MATLAB 主要由以下 5 部分组成：

1）MATLAB 语言体系：它是高层次的矩阵/数组语言，具有条件控制、函数调用、数据结构、输入输出、面向对象等程序语言特性。利用它既可以进行小规模编程，完成算法设计和算法实验的基本任务，也可以进行大规模编程，开发复杂程序。

2）MATLAB 工作环境：这是对 MATLAB 提供给用户的管理功能的总称，包括管理工作空间中的变量数据输入输出的方式和方法，以及开发、调试、管理点 M 文件的各种工具。

3）图像句柄系统：这是 MATLAB 图形系统的基础，包括完成 2D 和 3D 数据图示、图像处理、动画生成、图形显示等功能的高层 MATLAB 命令，以及开发图形用户界面（GUI）应用程序的各种工具。

4）MATLAB 数学函数库：对 MATLAB 使用的各种数学算法的总称，包括初等函数的算法及矩阵运算、矩阵分析等高层次数学算法。

5）MATLAB 应用程序接口（API）：这是 MATLAB 为用户提供的一个函数库，使得用户能够在 MATLAB 环境中使用 C 程序或 FORTRAN 程序，包括从 MATLAB 中调用自程序（DLL），读写 MAT 文件的功能。

MATLAB 是一个功能十分强大的系统，集数值计算、图形管理、程序开发为一体。另外，MATLAB 还具有很强的功能扩展能力，可以配备各种各样的工具箱以完成一些特定的任务，同时用户还可以根据自己的工作任务开发自己的工具箱。

10.1.1　数值计算

MATLAB 提供了几乎所有常用的初等函数和很多特殊的函数，还包括信号处理用的各种变换和滤波以及灵活的矩阵运算，可以说能够完成各种数值计算。用户可以通过编写 M 文件或调用已有的 M 文件来完成特殊功能的任务。在进行 MATLAB 程序设计时要注意以下几个问题：

1）MATLAB 的内部向量与矩阵运算速度比 MATLAB 编译解释的速度快，即为了提高运算效率，在程序设计中要尽可能使用向量和矩阵的运算，特别是在 for 或 while 等循环语句中尽可能转换成矩阵运算。例如：

```
>> t =0:.01 : 10; t(t= =0) =eps;
>> y =sin(t)./t;
```

要比用 for/while 循环语句实现快得多。

2）为将要使用的向量或矩阵预先分配内存空间。例如：

```
>> x =zeros(1, 1000);
>> x =[1 : 1000];
```

为 x 预先分配了内存空间而提高了运算速度。

3）减少变量屏幕显示和变量使用个数。

10.1.2　图形功能

MATLAB 的图形系统提供了用图形的方式表达数据的完美技术，这个系统建立在内部图形对象的管理技术之上。主要有以下几种绘制类型：

1）平面图形与坐标系：在 Windows 环境下，MATLAB 在图形对象（figure）窗口中绘制或输出图形。MATLAB 系统管理着图形窗口的生成、关闭和控制，能够同时生成多个图形窗口，每个图形窗口又可以有多个坐标系。MATLAB 中各种对象的层次关系如图 10-1 所示。

绘制 2D 图形的命令有多种：

- 向量式 plot（$\boldsymbol{v}$）：连接 $\{[i, \boldsymbol{v}(i)], i=1, 2, \cdots, n\}$ 生成一条曲线。
- 参数式 plot（$\boldsymbol{x}, \boldsymbol{y}$）：连接 $\{[\boldsymbol{x}(i), \boldsymbol{y}(i)], i=1, 2, \cdots, n\}$ 生成一条曲线。
- 矩阵式 plot（$\boldsymbol{A}_{m\times n}$）：以 1 到 m 为横坐标生成 n 条曲线。
- 混合式 plot（$\boldsymbol{X}, \boldsymbol{Y}$）：$\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 可以是向量或矩阵。

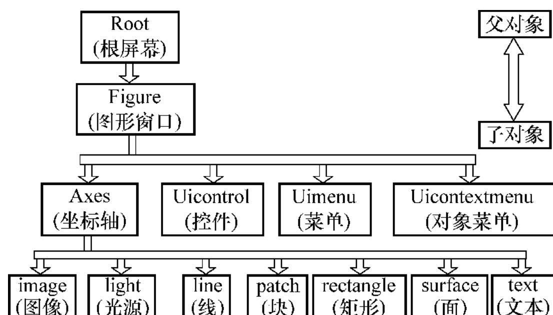


图 10-1　对象层次关系

2）三维图形：MATLAB 具有强大的三维图形功能，包括三维数据显示、空间曲线、曲面、分块、填充和旋转等功能和操作。3D 绘图命令有

- 向量式 plot（$\boldsymbol{x}$，$\boldsymbol{y}$，z）：连接$\{[\boldsymbol{x}(i), \boldsymbol{y}(i), z(i)], i=1, 2, \cdots, n\}$生成一条曲线。
- 网格曲面：mesh($\boldsymbol{X}$，$\boldsymbol{Y}$，$\boldsymbol{Z}$)。
- 实曲面：surf($\boldsymbol{X}$，$\boldsymbol{Y}$，$\boldsymbol{Z}$)。
- 等高线；contour3($\boldsymbol{X}$，$\boldsymbol{Y}$，$\boldsymbol{Z}$)。

3）彩色图形：通过色谱矩阵，MATLAB 可以生成全真彩色的各种彩色图形，如 surf($\boldsymbol{X}$，$\boldsymbol{Y}$，$\boldsymbol{Z}$，$\boldsymbol{C}$)，其中 $\boldsymbol{C}$ 是颜色矩阵。

4）图形图像控制：除了 MATLAB 图形系统的高层次操作，MATLAB 图形系统还具有底层的图形操作功能即通过图形句柄管理模式，按照图形对象来管理。每个图形对象都具有一定的属性，MATLAB 正是通过对属性的操作来控制和改变图形对象。

10.1.3 工具箱

MATLAB 已经成为一个系列产品：MATLAB 主包和各种工具箱（ToolBox）。功能丰富的工具箱将不同领域、不同方向的研究者吸引到 MATLAB 编程环境中来，成为 MATLAB 的忠实用户。功能型工具箱主要用来扩充 MATLAB 的符号计算功能、图形建模仿真功能、文字处理功能以及与硬件实时的交互功能，能用于多种科学学科，而领域型工具箱是专业性很强的工具箱如控制工具箱，信号处理工具箱等。下面介绍几个与本文有关的主要工具箱。

1）通信工具箱（Communication Toolbox）：提供了 100 多个函数和 150 多个 Simulink 模块用于通信系统的仿真和分析。按照模拟/数字通信系统模型，MATLAB 提供的函数分类见表 10-1。可由结构图直接生成可应用的 C 语言程序。

2）控制系统工具箱（Control System Toolbox）：提供了连续系统设计和离散系统设计，状态空间和传递函数，频域响应：博德图、奈奎斯特图、Nichols 图、时域响应、冲激响应、阶跃响应等。

表 10-1 通信系统工具箱的函数分类

过 程	相关的函数	过 程	相关的函数
信号发生器和显示设备	随机信号发生器、显示设备、误码率分析	调制和解调	带通模拟调制解调、基带模拟调制解调、带通数字调制解调、基带数字调制解调
信源编码	标量量化、A 律压缩 μ 律压缩、预测量化	多路复用	时分复用、频分复用、码分复用
纠错编码	线性分组码、海明码、循环码、BCH 码、RS 码、卷积码	发送/接收滤波器	余弦平方滤波器、希尔伯特滤波器、复合参数滤波器

3）图像处理工具箱（Image Processing Toolbox）：包含 100 多个工具函数，主要分成：图像类型转换、图像文件 I/O 和显示、图像的空间变换（插值缩放等）、图像数学变换（FFT、DCT 等）、线性滤波器设计及滤波、统计分析、图像增强、二值图像处理及彩色图像处理等。

4）信号处理工具箱（Signal Processing Toolbox）：覆盖了经典信号处理理论的大多数内容，不仅提供了大量的工具函数，而且还提供交互式的滤波器分析、设计工具和谱分析工具。函数主要分成以下 9 类：

① 信号和波形的产生；

② FFT、DCT、希尔伯特和其他一些数学变化；

③ 数字和模拟滤波其的设计、分析和实现，包括幅值和相位分析、零/极点分析，计算冲激响应和阶跃响应；

④ FFT、DCT、希尔伯特和其他一些数学变换，包括幅值和相位分析、零/极点分析，计算冲激响应和阶跃响应；

⑤ 随机信号处理与经典谱估计；

⑥ 多采样率信号处理；

⑦ 窗函数；

⑧ 时-频分析；

⑨ 倒谱分析。

高阶谱分析、小波分析和神经网络由于其本身发展很快，已经形成较为独立的学科体系，而其应用不仅仅限于信号处理，所以 MATLAB 提供了相应的专业工具箱，这些内容并没有包含到信号处理工具箱中。此外，数字图像虽然从本质上说是二维数字信号，但它具有一些特殊的性质，尤其是与人的感知紧密联系在一起，故也独立成为学科，MATLAB 也提供了专门的工具箱。

5）小波分析工具箱（Wavelet Toolbox）：小波分析作为一种强有力的信号分析工具，是由 Morlet 于 20 世纪 80 年代初在分析地球物理信号时提出来的，它是泛函分析、傅里叶变换、样条理论、调和分析、数值分析等多个学科相互交叉的结晶。它在语音信号处理、图像分析、地震信号分析、数据压缩等等许多领域得到广泛应用。小波分析是一种多尺度的信号分析方法，是分析非平稳信号的有力工具。它克服了短时傅里叶变换固定分辨率的弱点，既可以分析信号的概貌，又可分析信号细节。

MATLAB 小波分析工具箱集成了小波分析的许多研究成果，不仅提供了丰富的工具箱函数，而且提供了一个可视化得小波分析工具。按照功能划分，小波分析函数主要分成以下 7 类：

① 常用的小波基函数，如 morl、mexh、meyr、haar、dbN、symN、coifN 等；

② 连续小波变换及其应用，如 cwt；

③ 离散小波变换及其应用，如 dwt/idwt、dwt2/idwt2 ；

④ 小波包变换，如 wpdec/wprec、wpdec2/wprec2 ；

⑤ 信号和图像的多分辨率分解，如 wavedec/waverec、detcoef/appcoef ；

⑥ 基于小波的信号除噪，如 wden、wdencmp ；

⑦ 基于小波的信号压缩，如 wpdencmp 。

10. 2 Simulink 仿真

在实际工作中，一个大的应用系统往往很复杂，如果不借助专门的软件则很难把一个复杂模型输入给计算机从而对其进一步分析与仿真。Simulink 就是这样的一个通用软件包，它是实现动态系统模型、仿真的一个集成环境。它的主要功能是仿真（Simulink）和连接（Link），可方便地在模型窗口上建立系统模型，然后利用 Simulink 提供的功能对系统进行仿真。Simulink 的存在使 MATLAB 的功能进一步扩展，主要表现在：

1）实现了可视化建模，用户通过简单地鼠标操作就可以建立起直观的系统模型并进行仿真和分析。

2）实现了多工作环境间的文件互用和数据交换。

3）把理论研究和工程实现有机地结合在一起。

4）根据设计及使用的要求，对系统进行修改和优化以提高系统的性能，实现了高效开发系统的目标。

图 10-2 是 Simulink 库浏览器及工作空间。

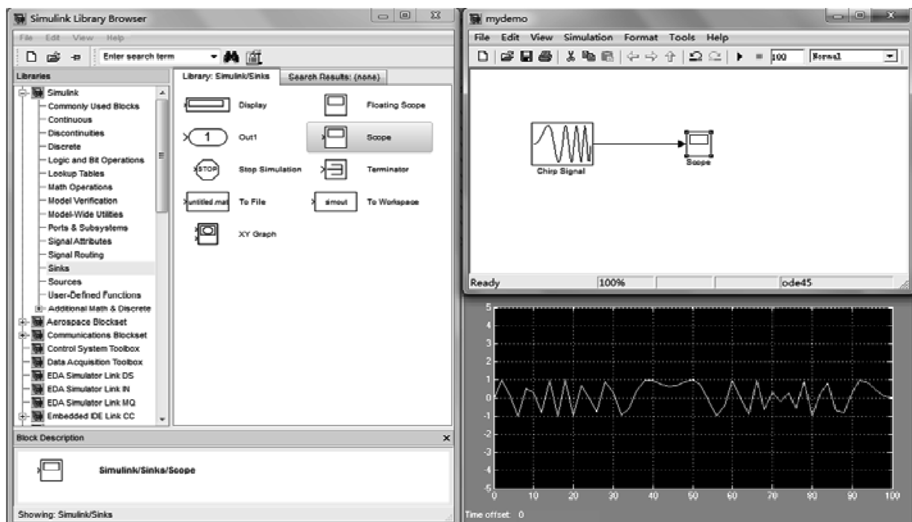


图 10-2 Simulink 库浏览器及建模

10.2.1　Simulink 简介

使用传统的方法实现系统的模拟和仿真，需要将系统的模块函数转化为微分或差分方程，再用某种程序语言编程进行仿真运算。Simulink 提供了图形化的用户界面，用户只需点击鼠标就可以轻易地完成模型的创建、调试和仿真工作，这样就大大地降低了仿真的难度。

Simulink 提供了大量的系统模块，包括信号、运算、显示和系统等多方面的功能，可以创建各种类型的仿真系统，实现丰富的仿真功能。用户也可以自己定义模块，进一步扩展模型的范围和功能以满足不同的需求。模型的创建与定义、模型的分析以及修正是使用 Simulink 的三大步骤，图 10-3 显示了典型的 Simulink 工作框图。

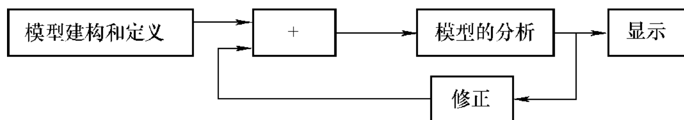


图 10-3　Simulink 操作框图

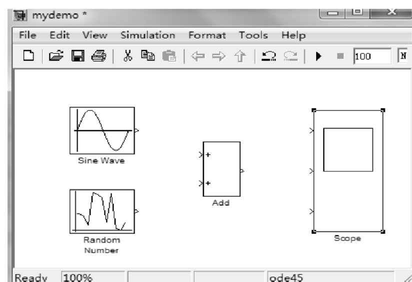
Simulink 具有相对独立的功能和使用方法，它支持线性和非线性系统、连续和离散时间系统，而且系统可以多进程。进行 Simulink 仿真的主要步骤如下：

1）运行 Simulink：在 MATLAB 命令窗口中直接键入 Simulink 并回车就会显示图 10-2 的 Simulink 启动界面。

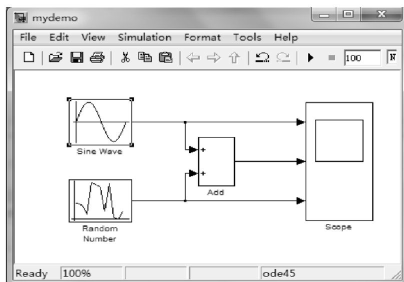
2）添加模块：用鼠标将所需的模块拖入新建模型窗口，如图 10-4a 所示。

3）编辑模块并连接成模型，还需对模块的参数进行设置以满足实验中所需要求。用鼠标双击模块，修改模块的参数或属性，如图 10-4b 与 c 所示。

4）进行系统仿真：在“Parameters”菜单中设置仿真的时间、步长以及算法等参数，如图 10-4d 所示，系统运行结果如图 10-5 所示。

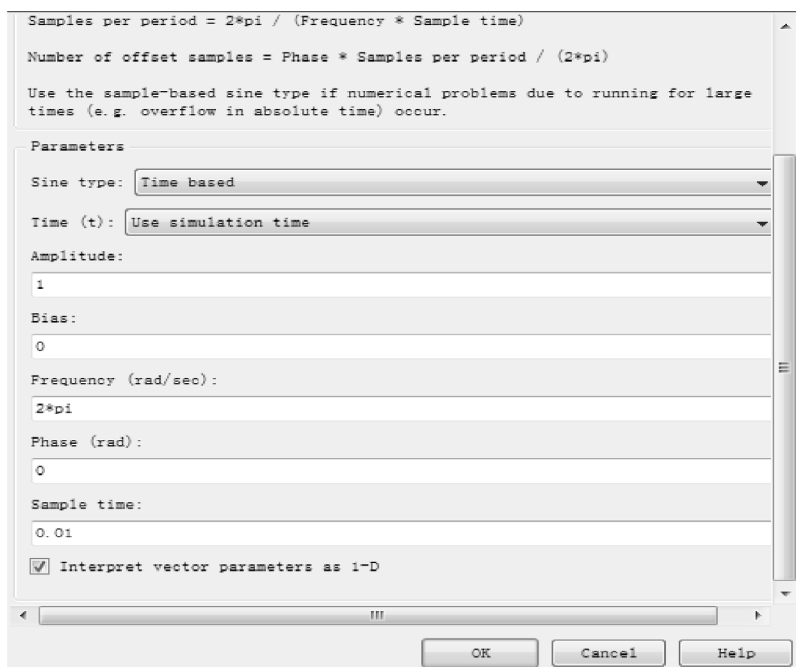


a) 在模型窗口中添加模块

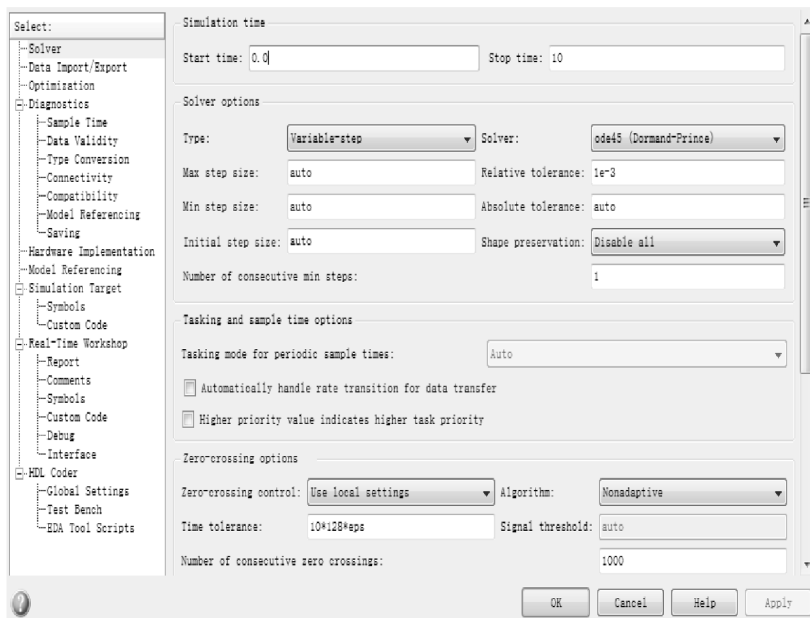


b)系统连线

图 10-4　Simulink 运行步骤



c) 正弦函数参数设置



d) 系统参数设置

图 10-4 Simulink 运行步骤（续）

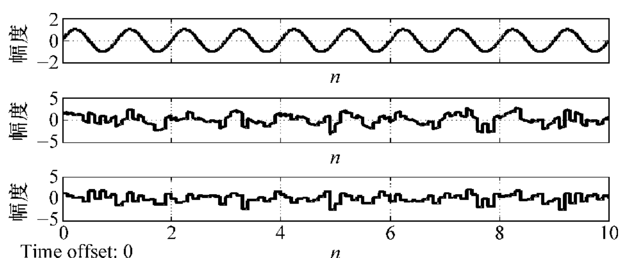


图 10-5　Simulink 运行结果

10.2.2　Simulink 中的白噪声分析

Simulink 中白噪声频带很宽，功率分布也非常均匀，在模拟仿真中作为噪声信号很具有代表性，如图 10-6a 所示。图中低通滤波器的截止频率为 0.3，带通滤波器的通带宽度为 0.6，高通滤波器的截止频率为 0.6，滤波器均为 32 阶带汉宁窗的 FIR 滤波器，运行结果如图 10-6b 所示。

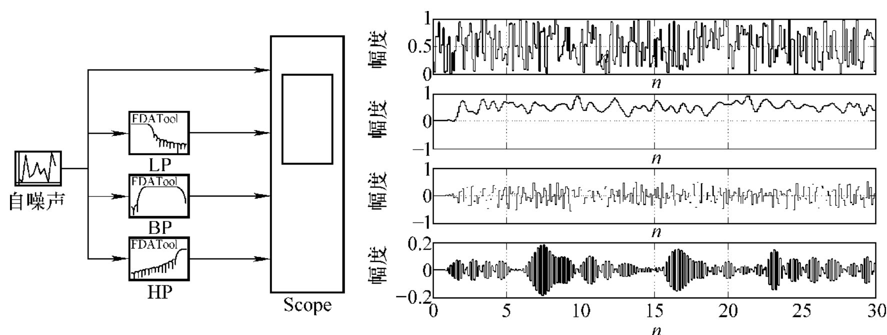


图 10-6　Simulink 白噪声分析

10.2.3　基于小波的数字信号处理

一维小波的信号除噪是小波分析的重要应用之一。在某些情况下，有用信号表现为低频信号或是一些比较平稳的信号，而噪声则表现为高频信号，所以除噪过程可按如下方法进行处理：首先对信号进行小波分解，采用 db4 进行 4 层分解如图 10-7 所示，则噪声部分通常包含在（1，1）、（2，1）、（3，1）和（4，1）中，因而可以以门限阈值等形式对小波系数进行处理，然后对信号进行重构，从而达到消除噪声

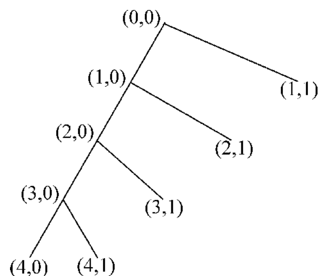


图 10-7　一维小波分解简图

的目的。图 10-8 是一维信号 ***s*** 按照 db2 小波 6 层分解的结果。一维信号小波除噪 Simulink 模型如图 10-9 所示。

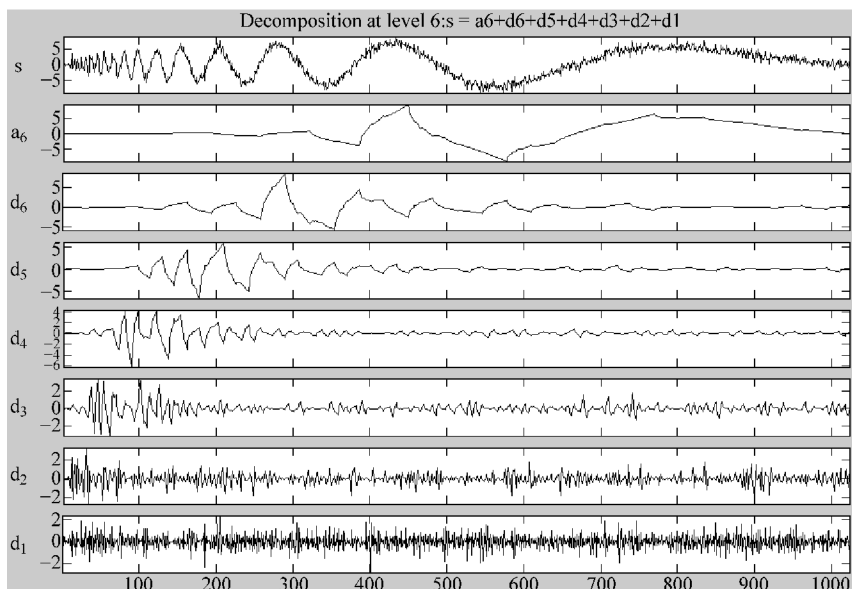


图 10-8 一维信号小波分解

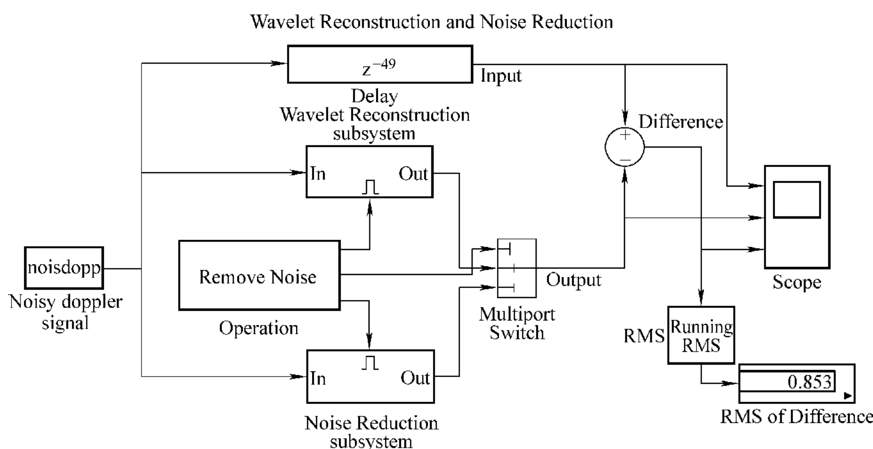


图 10-9 一维信号小波除噪 Simulink 模型

对信号除噪的目的就是抑制信号中的噪声部分，从而在信号中恢复有用信号。下面以正弦信号加白噪声为例进行除噪分析。

上述 Simulink 系统运行结果如图 10-10 所示。

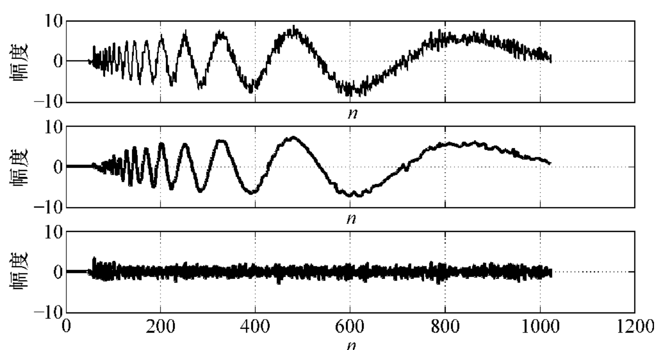


图 10-10　一维信号小波除噪

二维信号噪声的消除主要是图像的除噪，主要过程为：首先，二维信号的小波分解即选择合适的小波汉书和分解层次，然后计算信号到最后一层的分解；第二对高频系数进行阈值量化，即对每层的高频系数进行软阈值量化处理；第三，二维小波的重构，根据小波分解的最后一层的低频系数和经过修改的各层高频系数来计算二维信号的小波重构。系统运行结果如图 10-11 所示。

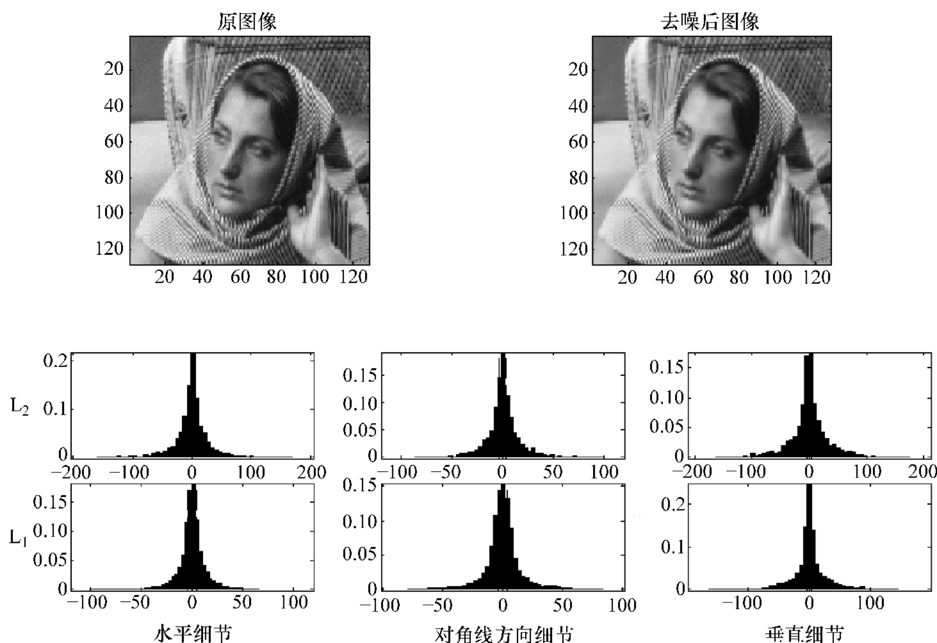


图 10-11　二维图像信号小波除噪

10.3 全相位滤波器仿真

基于 Simulink 建立 ap 滤波器结构，并验证与对应的 FIR 滤波器等效。经过封装的系统结构图如图 10-12 所示。

图 10-12 中输入是正弦波和高斯白噪声之和，正弦波频率为 5Hz，幅度为 1，噪声均值为 1，方差为 0，复合信号分别经过 ap 和等效的 FIR 滤波器。其中，ap 和 FIR 是经过封装的子系统，结构如图 10-13所示。

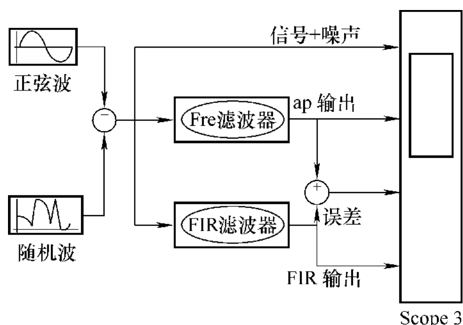
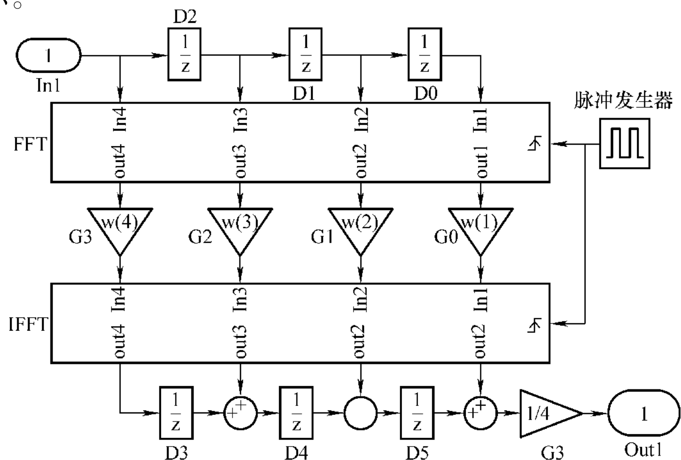
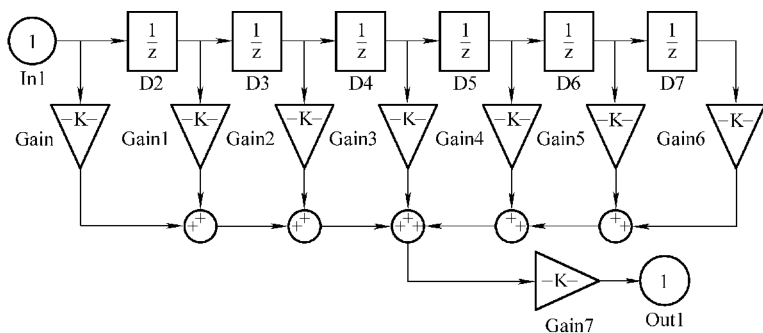


图 10-12 ap 验证系统结构



a) 4阶DFT ap系统结构



b) 等效FIR滤波器

图 10-13 ap 及等效 FIR 封装子系统

两个子系统模块设置如图 10-14 所示。系统运行结果如图 10-15 所示。

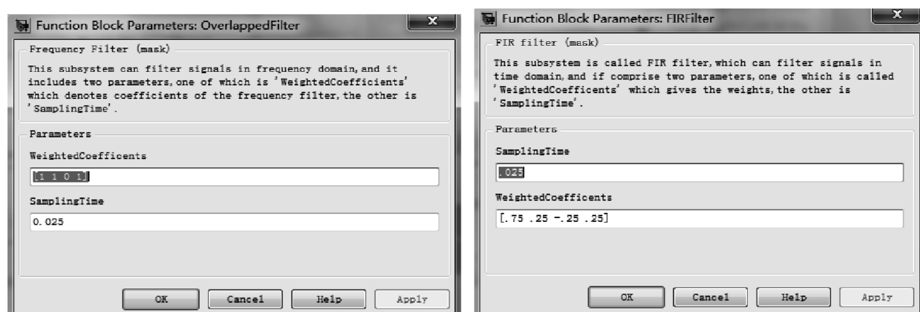
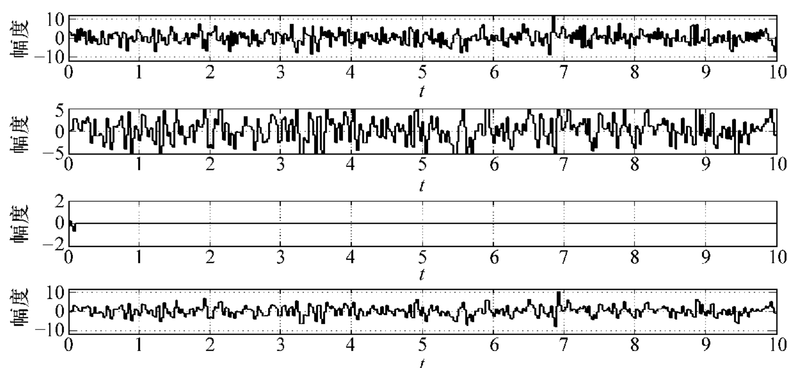
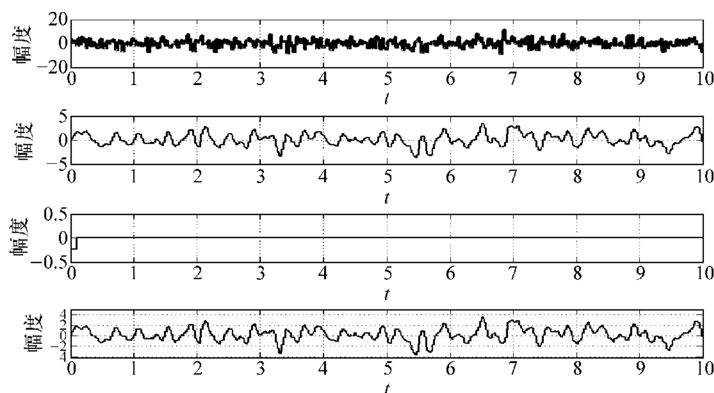


图 10-14　ap 及等效 FIR 子系统参数设置



a) $\boldsymbol{H}=[1\ 1\ 0\ 1]$



b) $\boldsymbol{H}=[1\ 0\ 0\ 0]$

图 10-15　ap Simulink 验证

上面实验结果证明了 ap 和 FIR 的等效性。利用 4 阶 ap 系统可以方便地通过级联方式进行扩展，例如 16 阶系统及各封装模块如图 10-16 所示。

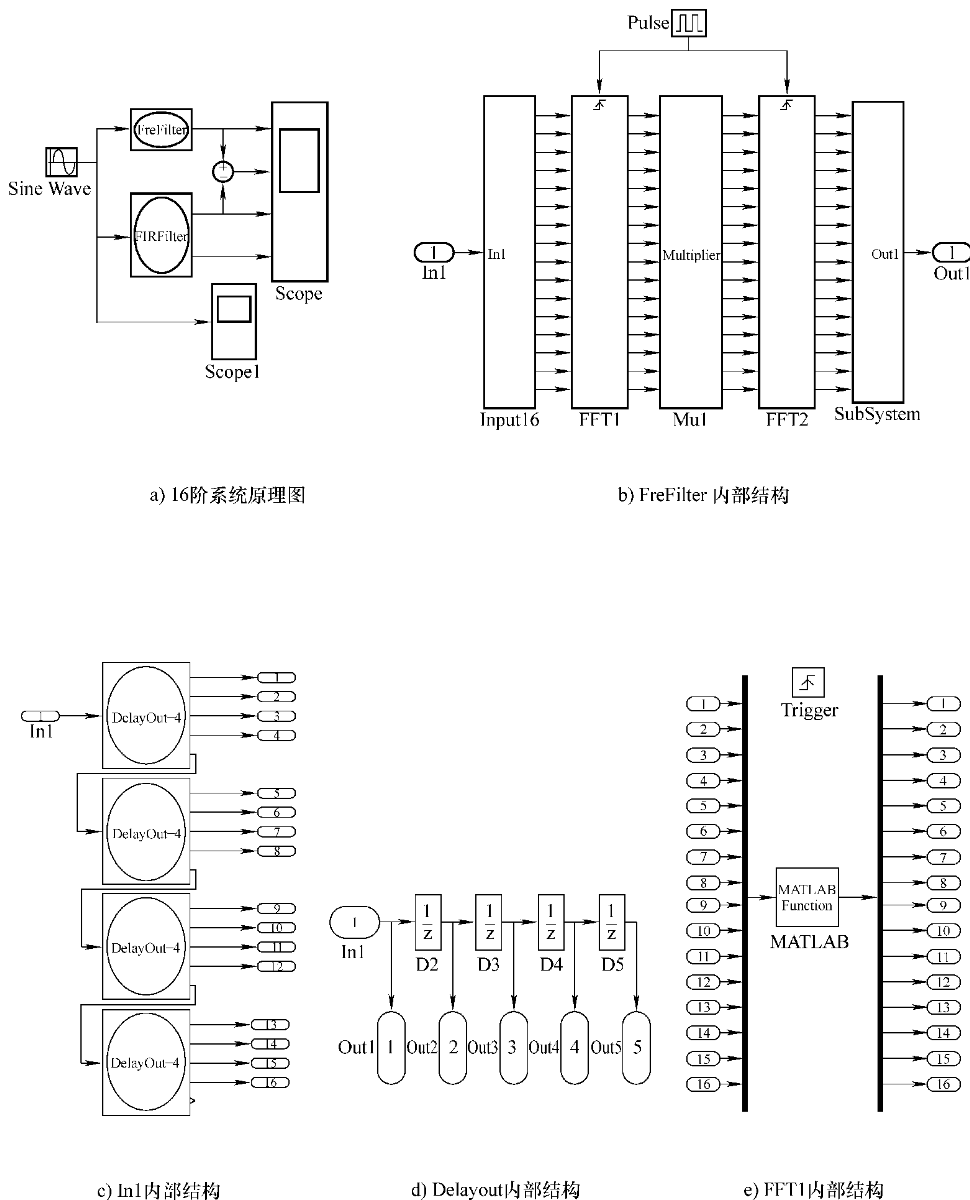
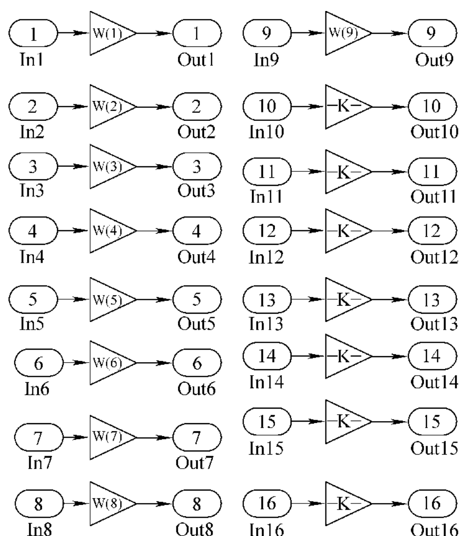
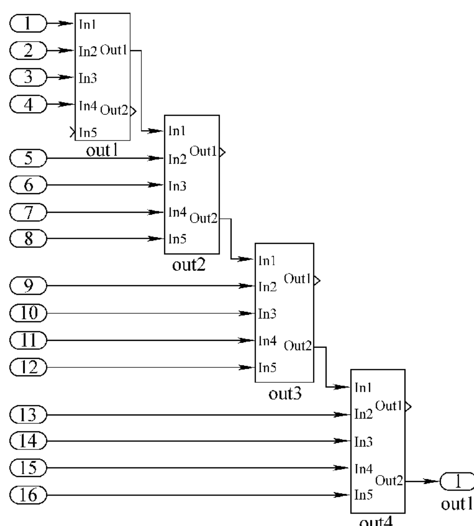


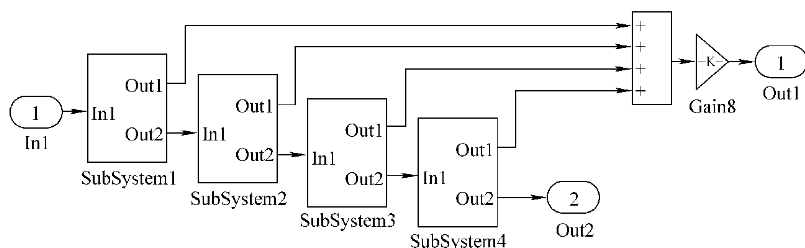
图 10-16　16 阶 ap Simulink 结构



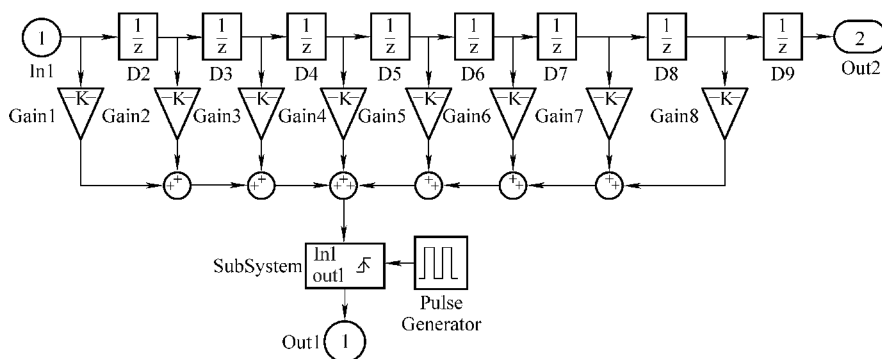
f) Multiplexer内部结构



g) out1内部结构



h) FIR 滤波器内部结构



i) FIR Filter内部结构

图 10-16　16 阶 ap Simulink 结构（续）

16 阶系统的仿真结果如图 10-17 所示。

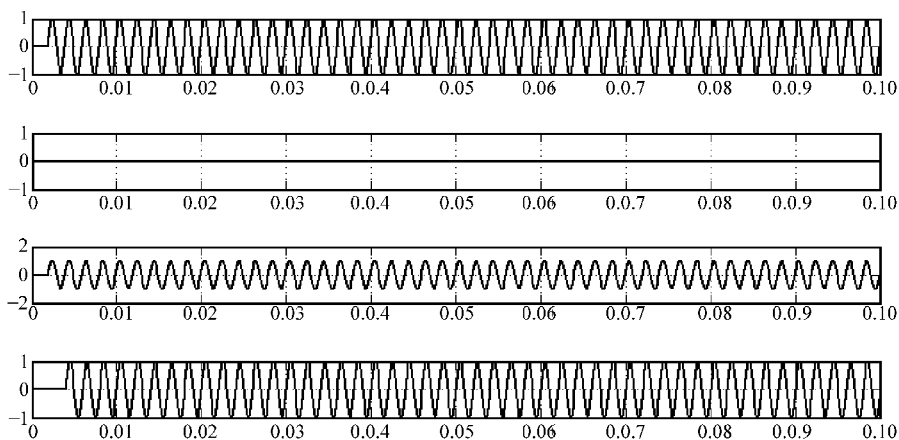


图 10-17　16 阶 ap 仿真结果

更高阶的 ap 系统可以按照上面方式来构造。上面的仿真系统实用于输入信号为实信号，若输入信号为复数信号，只需做少量的改动即可。

10.4　小结

本章首先介绍了目前使用十分广泛的专业性软件 MATLAB 的强大功能，并对 Simulink 模块的使用及 MATLAB 中小波功能和应用做了简单地说明；然后，基于 Simulink 设计 ap 仿真系统及设计理论推导得到的等效 FIR 系统，并对两者的等效性进行了验证；最后，在 4 阶 ap Simulink 结构的基础上通过子系统封装方式给出了 16 阶 ap 结构并仿真出实验结果。

本章只对 ap 结构的正确性进行了基础验证，而基于 ap 的各种应用的仿真工作比如二维 ap 系统设计、ap 模板设计、ap 谱分析等是今后需要进一步展开的工作。

附　　录

附录 A　矩阵谱证明

假定存在矩阵 $\boldsymbol{A}$ 和 $\boldsymbol{B}$，$\boldsymbol{A}$ 的维度是 $N/2\times N$，$\boldsymbol{B}$ 的维度是 $N\times N$ 且是酉矩阵。证明：矩阵 $\boldsymbol{ABA}^{\mathrm{H}}$的最大特征满足：$|\lambda_{\max}|\leqslant\max\limits_{1\leqslant i\leqslant N}\{h_i^2\}\cdot N$，其中 h_i是矩阵 $\boldsymbol{A}$ 的元素。

证明：矩阵 $\boldsymbol{A}$ 可以写成下式。

$$\boldsymbol{A}=\begin{bmatrix} 2h_0 & 2h_1 & 2h_2 & 2h_3 & \cdots & 2h_{\frac{N}{2}-1} & 2h_{\frac{N}{2}} & 2h_{\frac{N}{2}+1} & \cdots & 2h_{N-3} & 2h_{N-2} & 2h_{N-1} \\ 2h_0 & 4h_1 & 2h_2 & 2h_3 & \cdots & 2h_{\frac{N}{2}-1} & 2h_{\frac{N}{2}} & 2h_{\frac{N}{2}+1} & \cdots & 2h_{N-3} & 2h_{N-2} & 0 \\ 2h_0 & 4h_1 & 4h_2 & 2h_3 & \cdots & 2h_{\frac{N}{2}-1} & 2h_{\frac{N}{2}} & 2h_{\frac{N}{2}+1} & \cdots & 2h_{N-3} & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \\ 2h_0 & 4h_1 & 4h_2 & 4h_3 & \cdots & 4h_{\frac{N}{2}-1} & 2h_{\frac{N}{2}} & 0 & \cdots & 0 & 0 & 0 \end{bmatrix} \tag{A-1}$$

分解矩阵 $\boldsymbol{A}=[\boldsymbol{A}1,\ \boldsymbol{A}2]$，矩阵 $\boldsymbol{A}1$ 和 $\boldsymbol{A}2$ 定义如下：

$$\boldsymbol{A}_1=\begin{bmatrix} 2h_0 & 2h_1 & 2h_2 & 2h_3 & \cdots & 2h_{\frac{N}{2}-1} \\ 2h_0 & 4h_1 & 2h_2 & 2h_3 & \cdots & 2h_{\frac{N}{2}-1} \\ 2h_0 & 4h_1 & 4h_2 & 4h_3 & \cdots & 2h_{\frac{N}{2}-1} \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ 2h_0 & 4h_1 & 4h_2 & 4h_3 & \cdots & 4h_{\frac{N}{2}-1} \end{bmatrix} \quad \boldsymbol{A}_2=\begin{bmatrix} 2h_{\frac{N}{2}} & 2h_{\frac{N}{2}+1} & \cdots & 2h_{N-2} & 2h_{N-1} \\ 2h_{\frac{N}{2}} & 2h_{\frac{N}{2}+1} & \cdots & 2h_{N-2} & 0 \\ 2h_{\frac{N}{2}} & 2h_{\frac{N}{2}+1} & \cdots & 0 & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ 2h_{\frac{N}{2}} & 0 & \cdots & 0 & 0 \end{bmatrix} \tag{A-2}$$

令 $\boldsymbol{C}=\boldsymbol{ABA}^{\mathrm{H}}$，则可得到

$$\begin{aligned} \boldsymbol{C} &= [\boldsymbol{A}_1\ \boldsymbol{A}_2]\cdot\begin{bmatrix} \boldsymbol{B}_0 & \boldsymbol{B}_1 \\ \boldsymbol{B}_2 & \boldsymbol{B}_2 \end{bmatrix}\cdot\begin{bmatrix} \boldsymbol{A}_1^{\mathrm{H}} \\ \boldsymbol{A}_2^{\mathrm{H}} \end{bmatrix} \\ &= \boldsymbol{A}_1\boldsymbol{B}_0\boldsymbol{A}_1^{\mathrm{H}}+\boldsymbol{A}_2\boldsymbol{B}_2\boldsymbol{A}_1^{\mathrm{H}}+\boldsymbol{A}_1\boldsymbol{B}_1\boldsymbol{A}_2^{\mathrm{H}}+\boldsymbol{A}_2\boldsymbol{B}_3\boldsymbol{A}_2^{\mathrm{H}} \\ &= \boldsymbol{C}_1+\boldsymbol{C}_2+\boldsymbol{C}_3+\boldsymbol{C}_4 \end{aligned} \tag{A-3}$$

这样，就有

$$\begin{aligned}|\boldsymbol{C}-\lambda\boldsymbol{E}| &= |\boldsymbol{C}_1+\boldsymbol{C}_2+\boldsymbol{C}_3+\boldsymbol{C}_4-\lambda\boldsymbol{E}| \\ &= |(\boldsymbol{C}_1-\lambda_1\boldsymbol{E})+(\boldsymbol{C}_2-\lambda_2\boldsymbol{E})+(\boldsymbol{C}_3-\lambda_3\boldsymbol{E})+(\boldsymbol{C}_4-\lambda_4\boldsymbol{E})|\end{aligned} \tag{A-4}$$

如果以下表达式成立

$$|\boldsymbol{C}_1-\lambda_1\boldsymbol{E}|=0 \quad |\boldsymbol{C}_2-\lambda_2\boldsymbol{E}|=0 \quad |\boldsymbol{C}_3-\lambda_3\boldsymbol{E}|=0 \quad |\boldsymbol{C}_4-\lambda_4\boldsymbol{E}|=0 \tag{A-5}$$

则有

$$\begin{aligned}&\because |(C_1-\lambda_1\boldsymbol{E})+(C_2-\lambda_2\boldsymbol{E})+(C_3-\lambda_3\boldsymbol{E})+(C_4-\lambda_4\boldsymbol{E})| \\ &\leqslant |C_1-\lambda_1\boldsymbol{E}|+|C_2-\lambda_2\boldsymbol{E}|+|C_3-\lambda_3\boldsymbol{E}|+|C_4-\lambda_4\boldsymbol{E}|=0 \\ &\therefore |(\boldsymbol{C}_1-\lambda_1\boldsymbol{E})+(\boldsymbol{C}_2-\lambda_2\boldsymbol{E})+(\boldsymbol{C}_3-\lambda_3\boldsymbol{E})+(\boldsymbol{C}_4-\lambda_4\boldsymbol{E})|=0 \\ &\Rightarrow |\boldsymbol{C}-\lambda\boldsymbol{E}|=0 \Rightarrow \lambda=\lambda_1+\lambda_2+\lambda_3+\lambda_4\end{aligned} \tag{A-6}$$

因此 $\boldsymbol{C}$ 的特征值等于 $\boldsymbol{C}_1 \sim \boldsymbol{C}_4$ 特征值之和。另外，观察式（A-2），通过行变换，$\boldsymbol{A}_1$ 和 $\boldsymbol{A}_2$ 都可以转化成对角矩阵 $\boldsymbol{d}_1$ 和 $\boldsymbol{d}_2$。

$$\boldsymbol{d}_1=\begin{bmatrix}h_0 & 0 & \cdots & 0 \\ 0 & h_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & h_{\frac{N}{2}-1}\end{bmatrix} \quad \boldsymbol{d}_2=\begin{bmatrix}h_{\frac{N}{2}} & 0 & \cdots & 0 \\ 0 & h_{\frac{N}{2}+1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & h_{N-1}\end{bmatrix} \tag{A-7}$$

也就是说，存在可逆矩阵 $\boldsymbol{R}$ 使下式成立：

$$\boldsymbol{A}_1=\boldsymbol{R}\,\boldsymbol{d}_1 \Rightarrow \boldsymbol{C}_1=\boldsymbol{R}\,\boldsymbol{d}_1\boldsymbol{B}\,\boldsymbol{d}_1\,\boldsymbol{R}^H \Rightarrow \boldsymbol{C}_1 \cong \boldsymbol{d}_1\boldsymbol{B}\,\boldsymbol{d}_1 \tag{A-8}$$

式中，$\boldsymbol{B}_i(i=1\sim4)$ 是酉矩阵，即满足 $\boldsymbol{B}_i\boldsymbol{B}_i^{\mathrm{H}}=\boldsymbol{B}_i^{\mathrm{H}}\boldsymbol{B}_i=\boldsymbol{E}$。因此可对角化如下：

$$\boldsymbol{U}\,\boldsymbol{B}_i\,\boldsymbol{U}^{\mathrm{H}}=\mathrm{diag}\left\{b_0^{(i)} \quad b_1^{(i)} \quad \cdots \quad b_{\frac{N}{2}-1}^{(i)}\right\} \tag{A-9}$$

式中，$\boldsymbol{U}$ 是酉矩阵；$b_k^{(i)}(k=0,1,\cdots,N-1)$ 是 $\boldsymbol{B}_i$ 的特征值。以 $\boldsymbol{B}_0$ 为例，由式（A-6）和式（A-7）可得

$$\boldsymbol{C}_1 \cong \boldsymbol{d}_1\,\boldsymbol{U}^{\mathrm{H}}\mathrm{diag}(\boldsymbol{B}_0)\,\boldsymbol{U}\,\boldsymbol{d}_1 \tag{A-10}$$

矩阵与其转置矩阵相似，所以

$$\begin{aligned}&\boldsymbol{C}_1 \cong \boldsymbol{d}_1\mathrm{diag}(\boldsymbol{B}_0)\boldsymbol{d}_1=\mathrm{diag}\{h_0^2b_0^{(0)} \quad h_1^2b_1^{(0)} \quad \cdots \quad h_{\frac{N}{2}-1}^2b_{\frac{N}{2}-1}^{(0)}\} \\ &\Rightarrow \rho(\boldsymbol{C}_1)=\mathrm{diag}\left\{h_0^2b_0^{(0)} \quad h_1^2b_1^{(0)} \quad \cdots \quad h_{\frac{N}{2}-1}^2b_{\frac{N}{2}-1}^{(0)}\right\}\end{aligned} \tag{A-11}$$

同理可以得到

$$\begin{aligned}&\rho(\boldsymbol{C}_2)=\mathrm{diag}\left\{h_0h_{\frac{N}{2}}b_0^{(2)} \quad h_1h_{\frac{N}{2}+1}b_1^{(2)} \quad \cdots \quad h_{\frac{N}{2}-1}h_{N-1}b_{\frac{N}{2}-1}^{(2)}\right\} \\ &\rho(\boldsymbol{C}_3)=\mathrm{diag}\left\{h_0h_{\frac{N}{2}}b_0^{(1)} \quad h_1h_{\frac{N}{2}+1}b_1^{(1)} \quad \cdots \quad h_{\frac{N}{2}-1}h_{N-1}b_{\frac{N}{2}-1}^{(1)}\right\} \\ &\rho(\boldsymbol{C}_4)=\mathrm{diag}\left\{h_{\frac{N}{2}}^2b_0^{(3)} \quad h_{\frac{N}{2}+1}^2b_1^{(3)} \quad \cdots \quad h_{N-1}^2b_{\frac{N}{2}-1}^{(3)}\right\}\end{aligned} \tag{A-12}$$

$$\boldsymbol{B}=\begin{bmatrix}\boldsymbol{B}_0 & \boldsymbol{B}_1\\ \boldsymbol{B}_2 & \boldsymbol{B}_3\end{bmatrix}=\begin{bmatrix}\boldsymbol{B}_0 & 0\\ 0 & \boldsymbol{B}_3\end{bmatrix}+\begin{bmatrix}0 & \boldsymbol{B}_1\\ \boldsymbol{B}_2 & 0\end{bmatrix}=\boldsymbol{\alpha}+\boldsymbol{\beta}=\boldsymbol{\alpha}^H+\boldsymbol{\beta}^H \tag{A-13}$$

所以

$$\rho(\boldsymbol{C}) = [\rho(\boldsymbol{C}_1)\cup\rho(\boldsymbol{C}_4)]\cup[\rho(\boldsymbol{C}_2)\cup\rho(\boldsymbol{C}_3)]$$

$$\Rightarrow\max\{\rho(\boldsymbol{C})\} = \max\{\max\{\rho(\boldsymbol{C}_1)\cup\rho(\boldsymbol{C}_4)\},\ \max\{\rho(\boldsymbol{C}_2)\cup\rho(\boldsymbol{C}_3)\}\}$$

$$\max\{\rho(\boldsymbol{C}_1)\cup\rho(\boldsymbol{C}_4)\} < \max_{0\leqslant i\leqslant N-1}\{h_i^2\}\cdot\left(\sum_{j=0}^{\frac{N}{2}-1}|b_j^{(0)}|+\sum_{j=0}^{\frac{N}{2}-1}|b_j^{(3)}|\right)$$

$$\max\{\rho(\boldsymbol{C}_2)\cup\rho(\boldsymbol{C}_3)\} < \max_{0\leqslant i\leqslant\frac{N}{2}}\{|h_i h_{i+\frac{N}{2}}|\}\cdot\left(\sum_{j=0}^{\frac{N}{2}-1}|b_j^{(1)}|+\sum_{j=0}^{\frac{N}{2}-1}|b_j^{(2)}|\right)$$

$$\left(\sum_{j=0}^{\frac{N}{2}-1}|b_j^{(0)}|+\sum_{j=0}^{\frac{N}{2}-1}|b_j^{(3)}|\right)^2\leqslant 2\left(\sum_{j=0}^{\frac{N}{2}-1}|b_j^{(0)}|^2+\sum_{j=0}^{\frac{N}{2}-1}|b_j^{(3)}|^2\right)$$

$$\tag{A-14}$$

$$\left(\sum_{j=0}^{\frac{N}{2}-1}|b_j^{(1)}|+\sum_{j=0}^{\frac{N}{2}-1}|b_j^{(2)}|\right)^2\leqslant 2\left(\sum_{j=0}^{\frac{N}{2}-1}|b_j^{(1)}|^2+\sum_{j=0}^{\frac{N}{2}-1}|b_j^{(2)}|^2\right)$$

$$\boldsymbol{\alpha} = \boldsymbol{\alpha}^H\ \boldsymbol{\beta} = \boldsymbol{\beta}^H$$

$$\sum_{j=0}^{\frac{N}{2}-1}|b_j^{(0)}|^2+\sum_{j=0}^{\frac{N}{2}-1}|b_j^{(3)}|^2 = \|\boldsymbol{\alpha}\|_2^2\quad \sum_{j=0}^{\frac{N}{2}-1}|b_j^{(1)}|^2+\sum_{j=0}^{\frac{N}{2}-1}|b_j^{(2)}|^2 = \|\boldsymbol{\beta}\|_2^2$$

由此得到如下结论：

$$\max\{\rho(C)\}<\max\left\{\max_{0\leqslant i\leqslant N-1}\{h_i^2\}\cdot(2\ \|\alpha\|_2^2)^{1/2}\quad \max_{0\leqslant i\leqslant\frac{N}{2}}\{|h_i h_{i+\frac{N}{2}}|\}\cdot(2\ \|\beta\|_2^2)^{1/2}\right\} \tag{A-15}$$

对于单位冲激响应矩阵 $\boldsymbol{h}$ 和余弦矩阵 $\boldsymbol{C}$，结合上式（A-15）可得

$$\max\{\rho(\boldsymbol{C})\}<\max_{1\leqslant i\leqslant N-1}\{h_i^2\}\cdot\left(2\cdot\frac{N^2}{2}\right)^{1/2}=\max_{1\leqslant i\leqslant N-1}\{h_i^2\}\cdot N \tag{A-16}$$

附录 B　多相分解

多相分解是多率信号处理中的一种基本方法，在实现滤波器组时非常有用。多相分解是指将数字滤波器的传输函数 $H(z)$ 分解成不同相位的组合。

1. **I** 型多相分解

设 $h(n)$ 为某离散系统的单位脉冲序列，$H(z)$ 是其系统函数，令：

$$e_n(k)=h(kM+n),\ n=0,1,\cdots,M-1 \tag{B-1}$$

称 $e_n(k)$ 为 $h(k)$ 第 n 个多相分量。若 $e_n(k)$ 的 Z 变换为 $E_n(z)$，则称 $E_n(z)$ 为 $H(z)$ 的第 n 个多相分量。显然，$H(z)$ 可以有 M 个多相分量。推导如下：

$$\begin{aligned} H(z) &= \sum_{k=-\infty}^{+\infty} h(k)z^{-k} = \sum_{n=0}^{M-1}\sum_r h(rM+n)z^{-(rM+n)}k = rM+n,n \\ &= 0,1,\cdots M-1,r\in Z \\ &= \sum_{n=0}^{M-1}\sum_r [e_n(r)z^{-rM}]z^{-n} \\ &= \sum_{n=0}^{M-1} E_n(z^M)z^{-n} \end{aligned} \tag{B-2}$$

2. **II** 型多相分解

根据 I 型多相分解，记 $R_n(z)=E_{M-1-n}(z)$ 并代入式（B-2）得到 II 型分解表达式为

$$H(z)=\sum_{n=0}^{M-1} R_n(z^M)z^{-(M-1-n)} \tag{B-3}$$

3. 多相分解实现的 FIR 结构（见图 B-1）

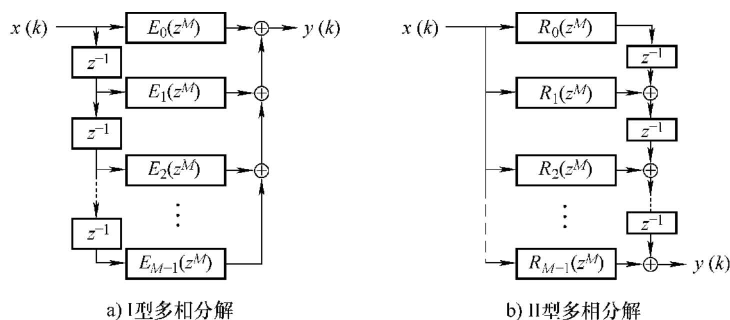


图 B-1　多相分解的 FIR 实现

4. 多相分解实现抽取滤波器

$h(n)$ 和 $h(nM+k)$ 关系如图 B-2 所示。



图 B-2　$h(n)$ 和 $h(nM+k)$ 的关系

多相分解实现的抽取器如图 B-3 所示。

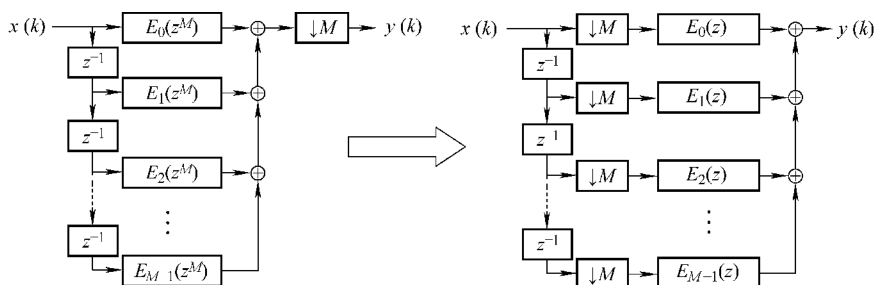


图 B-3　多相分解实现的抽取器

多相分解实现的内插器如图 B-4 所示。

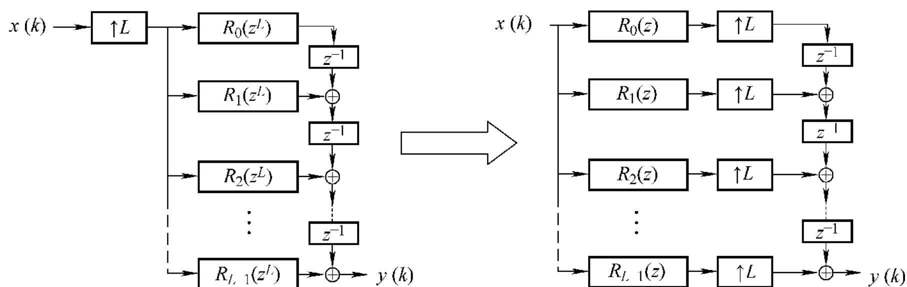


图 B-4　多相分解实现的内插器

图 B-5 所示系统为抽取和内插处理。

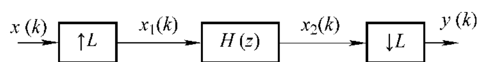


图 B-5　抽取和内插处理

由图 B-5 可得：

$$H(z)=\sum_{n=0}^{M-1}E_n(z^M)z^{-n},X_1(z)=X(z^M),X_2(z)=H(z)X_1(z)=\sum_{n=0}^{M-1}E_n(z^M)z^{-n}X(z^M)$$

$$\Rightarrow Y(z)=\frac{1}{M}\sum_{l=0}^{M-1}X_2(z^{1/M}W_M^l)=\frac{1}{M}\sum_{l=0}^{M-1}\sum_{n=0}^{M-1}X(z)E_n(z)z^{-n/M}W_N^{-nl}$$

$$=\sum_{n=0}^{M-1}X(z)E_n(z)z^{-n/M}\delta_M(n)=X(z)E_0(z)$$

（B-4）

附录 C　几种特殊的滤波器

1. 半带滤波器

半带滤波器的单位脉冲响应满足：

$$h(2n)=\begin{cases}c & n=0\\0 & n\neq 0\end{cases}\tag{C-1}$$

半带滤波器的多相域特性如下：

$$H(z)=c+z^{-1}E_1(z^2)\tag{C-2}$$

半带滤波器的 z 域特性如下：

$$H(z)+H(-z)=2c\tag{C-3}$$

如果 $h(k)$ 是实偶对称，则有：

$$H(-e^{j\omega})=H[e^{j(\omega-\pi)}]=H[e^{j(\pi-\omega)}]\Rightarrow H(e^{j\omega})+H[e^{j(\pi-\omega)}]=1,\ 取\ c=0.5\tag{C-4}$$

当 $\omega=\pi/2$ 时，由式（C-4）易知：

$$H(e^{j\pi/2})=0.5\tag{C-5}$$

若 $H(e^{j\omega_p})=1-\delta$，则 $H[e^{j(\pi-\omega_p)}]=1-H(e^{j\omega_p})=\delta=H(e^{j\omega_s})$，从而得如下结论：

$$\omega_p+\omega_s=\pi\tag{C-6}$$

如图 C-1 所示为典型的半带滤波器频域特性。

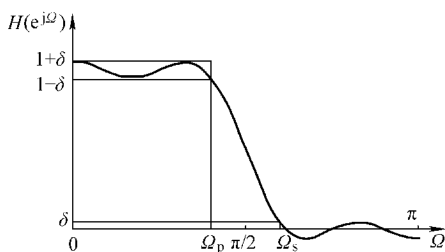


图 C-1　半带滤波器频域特性

设半带滤波器最高非零系数为 $h(k)$，显然 $k=2j+1$，则实系数零相位半带 FIR 滤波器长度满足：

$$N=2K+1=4J+3\tag{C-7}$$

2. 互补传递函数

（1）严格互补特性

如果传递函数 $H_0(z)$，$H_1(z)$，…，$H_{M-1}(z)$ 满足：

$$\sum_{k=0}^{M-1} H_k(z) = cz^{-n_0}，c\text{ 是非零常数} \tag{C-8}$$

则称这组传递函数是严格互补的。

（2）功率互补特性

如果传递函数 $H_0(z)$，$H_1(z)$，…，$H_{M-1}(z)$ 满足：

$$\sum_{k=0}^{M-1} |H_k(z)|^2 = c，c\text{ 是非零常数} \tag{C-9}$$

则称这组传递函数是功率互补的。

（3）全通特性

如果传递函数 $H_0(z)$，$H_1(z)$，…，$H_{M-1}(z)$ 满足：

$$\sum_{k=0}^{M-1} H_k(z) = A(z)，A(z)\text{ 是全通函数} \tag{C-10}$$

则称这组传递函数是全通互补的。

参 考 文 献

［1］王兆华．二维列率重叠数字滤波器［J］．电子学报，1985，13（6）：13-18.

［2］王兆华，H. Amiri. 二维低通重叠数字滤波器的研究［J］．通信学报，1985，13（6）：13-18.

［3］王兆华，H. Amiri. 二维抽取和内插［J］．信号处理，1987，3（4）：215-221.

［4］王兆华．重叠并元卷积［J］．信号处理，1994，10（1）：29-35.

［5］王兆华．计算机图像处理方法［M］．北京：中国宇航出版社，1993.

［6］侯正信．离散余弦列率滤波器的设计和应用［J］．天津大学学报，1999，32（3）：324-328.

［7］侯正信，刘建忠，宋占杰，等．全相位多维多抽样率滤波器设计［J］．天津大学学报，2011，44（4）：331-338.

［8］侯正信，胡莹莹，杨爱萍，等．全相位 DCT 自适应算法［J］．计算机工程与应用，2012，48（6）：162-165.

［9］李莉，侯正信，王成优，等．基于全相位 DCT/IDCT 内插的去马赛克算法［J］．光电工程，2008，35（12）：96-100.

［10］王兆华，黄翔东．数字信号全相位谱分析与滤波技术［M］．北京：电子工业出版社，2009.

［11］黄晓红．全相位数字间断信号处理［D］．天津：天津大学，2006.

［12］徐妮妮．全相位 FIR 滤波器组［D］．天津：天津大学，2005.

［13］徐妮妮，侯正信．全相位半带滤波器及其应用［J］．天津大学学报，2005，38（3），206-211.

［14］侯正信．全相位列率滤波器设计和实现［J］．信号处理，2001 年增刊，Vol.（17）：132-135.

［15］王兆华，侯正信，苏飞．全相位数字滤波［J］．信号处理，2003 年增刊，Vol.（19）：1-4.

［16］王兆华，侯正信，苏飞．全相位 FFT 频谱分析［J］．通信学报，2003，24（11A）：6-19.

［17］侯正信．三种二维重叠数字滤波器的构造［J］．天津大学学报，1985，Vol.（1）：29-72.

［18］侯正信，刘建忠，宋占杰，杨爱萍．全相位多维多抽样率滤波器设计［J］．天津大学学报，2011，44（4）：331-338.

［19］Winrow，B.，Steans，S. D. 自适应信号处理［M］．王永德，龙宪惠，译．北京：机械工业出版社，2008.

［20］黄翔东．全相位数字信号处理［D］．天津：天津大学，2006.

［21］姚庆栋，毕厚杰，王兆华，徐梦侠，等．图像编码基础［M］．北京：清华大学出版

社，2006.
[22] 丁玉美，高西全，彭学愚．数字信号处理［M］．西安：西安电子科技大学出版社，1995.
[23] 吴大正，杨林耀，张永瑞．信号与线性系统分析［M］．北京：高等教育出版社，2005.
[24] 崔锦泰．小波分析导论［M］．西安：西安交通大学出版社，1997.
[25] 熊洪允，曾绍标，毛云英．下册：应用数学基础［M］．天津：天津大学出版社，1998.
[26] 李海涛，邓樱．MATLAB 程序设计教程［M］．北京：高等教育出版社，2013.
[27] 伯晓晨，李涛，刘路．MATLAB 工具箱应用指南——信息工程篇［M］．北京：电子工业出版社，2000.
[28] 苏飞．带窗全相位数字滤波器设计及应用研究［D］．天津：天津大学，2003.
[29] 苏飞，王兆华．基于变换域全相位 FIR 自适应滤波算法［J］．电子学报，2004，32（11）：1859-1863.
[30] 苏飞，王兆华．一种新的带窗重叠自适应滤波器［J］．电子与信息学报，2005，27（1）：27-30.
[31] 苏飞，王兆华．基于矩形二叉分解的窗函数设计［J］．天津大学学报，2004，20（4）：1-4.
[32] 苏飞，王兆华．二维 FOURIER 带窗内插模板的设计［J］．中国图像图形学报，2004，9（4）：439-444.
[33] 苏飞，王兆华．全相位 FIR 滤波器及其在频谱分析中的应用［J］．数据采集与处理，2004，19（1）：60-67.
[34] 苏飞，王兆华．DCT 域全相位 FIR 滤波器设计和实现［J］．天津大学学报，2004，37（12）：1110-1114.
[35] 苏飞，王兆华．DFT 域全相位 FIR 滤波器设计和实现［J］．信号处理，2004，37（12）：1110-1114.
[36] 苏飞，王兆华．一种新结构频率域重叠数字滤波器［J］．电路与系统学报，2004，9（1）：46-72.
[37] 苏飞，王兆华．基于带双窗重叠域 FIR 自适应除噪［J］．电子测量与仪器学报，2004，18（4）：769-773.
[38] 李志国．全相位 DCT 数据内插方法的应用研究［D］．天津：天津大学，2009.
[39] 苏飞，卢晋，郝芸．基于全相位滤波器的 3D 图像恢复处理算法［J］．天津理工大学学报，2014，30（1）：40-46.